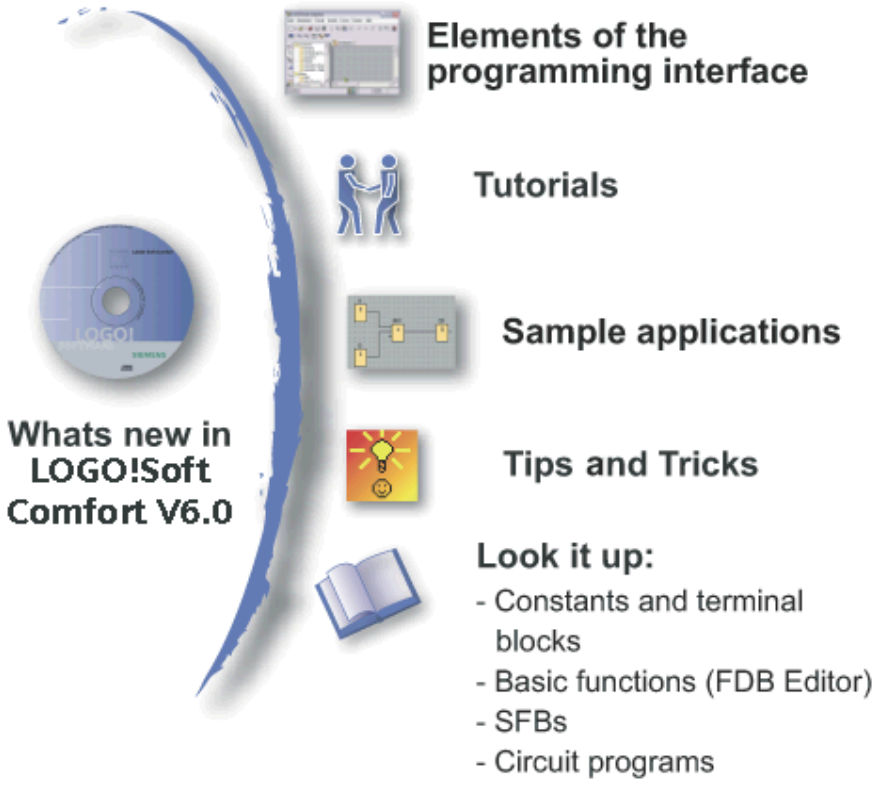
Contents

1 LOGO!Soft Comfort V6.0

1.1 Welcome to LOGO!Soft Comfort V6.0!



Help -> Content

Help -> Context-sensitive help

1.2 CD contents

The CD ROM

The CD-ROM included with your installation software for LOGO!Soft Comfort V6.0 contains additional useful information.

Start.html

This file guides you through the contents of the CD-ROM. It helps you to perform these tasks:

- Install LOGO!Soft Comfort
- Start LOGO!Soft Comfort

The file also provides you with access to many more items, including the following items:

- CAD drawings
- Manuals
- Drivers

Folders on the CD-ROM:

The list below describes some of the folder contents of the CD_ROM:

- The ..\CAD folder contains CAD drawings of the LOGO! devices.
- The..\Manuals folder contains the current LOGO! manual as a PDF file that you can read with AcrobatReader.
- The \Onlinehelp folder contains the current LOGO!Soft Comfort online help as a PDF file that you can read with AcrobatReader.
- The \Readme folder contains an HTML readme file that includes information that you need to know prior to installation.
- The ..\Sample folder contains a few sample applications, which give you example solutions for a few of the many fields of applications of the versatile LOGO!.
- The ..\Windows\Tools\Acrobat folders contain the Adobe AcrobatReader, which you need to view and print the PDF files of the manual and online help.
- The ..\Linux\Tools\Application, ..\Mac\Tools\Application and ..\Windows\Tools\ Application folders on the full version CD-ROM contain installation versions of LOGO!Soft Comfort for the named operating systems. As an alternative to installing LOGO!Soft Comfort, simply copy the corresponding ..\Tools\Application folder to your hard disk drive and start LOGO!Soft Comfort by calling ..\Application\LOGOComfort.

1.3 What's new in LOGO!Soft Comfort V6.0?

New SFBs

- Pulse Width Modulator (PWM)
- Analog Math

Updated SFBs

The following SFBs support new features:

- Message text
- Operating hours counter
- Weekly timer
- Yearly timer
- Analog watchdog
- Up/down counter

Additionally, LOGO!Soft Comfort supports new reference parameters for many blocks.

New LOGO! TD module

LOGO! supports a new text display module. This LOGO! TD module extends the display and user interface capabilities of LOGO! Basic. LOGO!Soft Comfort provides configuration of the following LOGO! TD features:

- Power-on screen
- Function keys
- Message texts
- Backlight function

For further information about the LOGO! TD module, refer to your product information and the LOGO! manual.

New features

LOGO!Soft Comfort V6.0 together with the LOGO! 0BA6 hardware series provides the following new features:

- Method to delete user program and password from LOGO! 0BA6
- Additional languages, resolution, and backlight function for the LOGO! 0BA6 display
- Ability to perform online tests of LAD circuit programs
- Display of PI controller analog output value in a trend view during simulation or online test
- Modem communication between a PC and LOGO! 0BA6
- USB cable communication between a PC and LOGO! Basic module
- New memory card, battery card and combined memory/battery card for LOGO! 0BA6 devices.

Changed functionality

LOGO!Soft Comfort V6.0 provides changes in the following areas:

- Number of user program blocks increased to 200 blocks
- Additional retentive memory for a total of 250 bytes
- Configuration support for all I/O changes to LOGO! basic and expansion modules

Prior versions

For information about features that were new in versions prior to LOGO!Soft Comfort V6.0, refer to the documentation for earlier versions. Current and previous documentation can be found at the LOGO! Internet web site.

1.4 Ladder Diagram (LAD) and Function Block Diagram (FBD)

LOGO!Soft Comfort provides you with two options of creating circuit programs:

- Ladder diagram (LAD)
- Function block diagram (FBD)

Who will use the ladder diagram (LAD)?

Users who are used to working with circuit diagrams.

Who will use the function block diagram (FBD)?

Users who are familiar with the logic boxes of Boolean algebra.

Differences between LAD and FBD in the online help

The online help primarily describes the FBD Editor, because its functionality is closely related to that of the LAD Editor. Where LAD differs, the online help will describe the differences. The following symbol in the online help indicates a difference in LAD and FBD functionality:

Converting circuit programs

Information on the conversion of circuit programs from LAD to FDB is found here.

Information on the conversion of circuit programs from FBD to LAD is found here.

Switching between LAD and FDB

Information on this topic is found here.

1.5 LOGO! and LOGO!Soft Comfort on the Internet

http:/www.siemens.com/logo/

From the Support selection at this Internet URL you will find abundant information about LOGO! and LOGO!Soft Comfort:

- Updates and upgrades for LOGO!Soft Comfort
- Further language packages, if the LOGO!Soft Comfort CD-ROM does not contain your language
- Numerous sample programs and applications
- FAQs (frequently asked questions)
- Downloads of current manuals and training documentation
- News and other information

You are welcome to visit this site!

1.6 Compatibility

Compatibility with previous LOGO! Hardware series

LOGO!Soft Comfort V6.0 is optimized for LOGO! devices of the 0BA6 series (indicated by the order number).

You can also use the current LOGO!Soft Comfort version to create circuit programs for the previous LOGO! hardware series. You cannot, however, download programs that use the new SFBs or new SFB parameters to LOGO! devices prior to the 0BA6 series. LOGO!Soft Comfort provides a list of hardware that is compatible with your circuit program from the Tools -> Select Hardware menu command. You can download your circuit program to any device in the list.

LOGO!Soft Comfort V6.0 adds reference functionality to many existing SFBs. You cannot download programs that use the new reference parameters to LOGO! devices prior to the 0BA6 series.

You can continue to use the input connectors of some of the LOGO! 0BA6 devices as they were prior to 0BA6, that is as inputs. However with the 0BA6 series, these inputs can also be used for analog inputs or high-speed counters. Existing LOGO!Soft Comfort programs that use these connectors as inputs will function in the same way as they did with the 0BA5 and earlier series. New programs can make use of the new analog inputs and high-speed counters. Refer to your product information and the LOGO! manual for detailed information about these modules: LOGO!12/24RC, LOGO!12/24RCo, LOGO!24, and LOGO!24o.

Any differences concerning the operation of LOGO!Soft Comfort that are based on differences between the previous series and the current LOGO! series 0BA6 are described separately. If programming differences exist based on the LOGO! series, this help system uses this graphic to alert you to those differences:

Compatibility with previous versions of LOGO!Soft Comfort

You can edit and expand circuit programs written with older versions of LOGO!Soft Comfort using your current LOGO!Soft Comfort version.

Compatibility with previous LOGO! memory cards

Refer to the LOGO! manual for compatibility information regarding LOGO! memory cards. LOGO!Soft Comfort does not access programs stored on memory cards).

LOGO! Devices with AS-Interface

You can connect the modular LOGO! via a communication module to an AS-Interface bus.

In doing so, the AS inputs and outputs behave just like standard inputs and outputs.

See also

AS-Interface

Here youfind important information about LOGO! devices with integrated AS-Interfaces.

LOGO! Hardware

Here youfind information about the individual hardware series. This also includes a table from which you can see which basic and special functions are available as of which hardware series.

1.7 LOGO! with AS Interface

1.7.1 AS-Interface

Contents

Here youdiscover what you must note when using a LOGO! with integrated AS-Interface.

Converting circuit programs

Circuit programs containing AS-Interface inputs or AS-Interface outputs which were created for previous versions of the modular LOGO! are converted in the following operations:

- When the circuit program is assigned to a modular LOGO! via Tools -> Select Hardware
- When the circuit program is downloaded without changes to a modular LOGO!

If one or more AS-Interface inputs or outputs are cut out of an older circuit program and pasted into the circuit program for a modular LOGO!, the inserted I/Os are also converted in the new circuit program.

Conversion rules

The inputs Ia1 to Ia4 are converted to I13 to I16. If the target inputs for the conversion are occupied otherwise, the source is converted to the next free input with the lowest block number.

Outputs Qa1 to Qa4 are converted to Q9 to Q12. If the target outputs for the conversion are occupied otherwise, the source is converted to the next free output with the lowest block number.

After conversion, the Info window shows you which AS-Interface I/Os were converted to I/Os of the modular LOGO!. If the conversion is not compatible to your physical hardware structure, you must adapt the block numbers of the relevant I/Os via the block properties dialog.

Additional constants and connecting terminals

Note that LOGO! versions 0BA0 to 0BA2 do not distinguish between standard inputs and AS-Interface inputs. Due to the modular structure of the devices as of version 0BA3, the block number of a digital, analog or AS-Interface block is determined by the slot position of the expansion module.

AS-Interface inputs

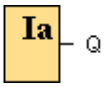
The relevant LOGO! versions are also equipped with Ia inputs used for the ASi bus.

Modular LOGO! devices as of the series 0BA3 do not distinguish between normal inputs and AS-Interface inputs. The user determines the type of input based on the modules used and the order in which they are installed. This is why AS-Interface inputs are here displayed only as I input.

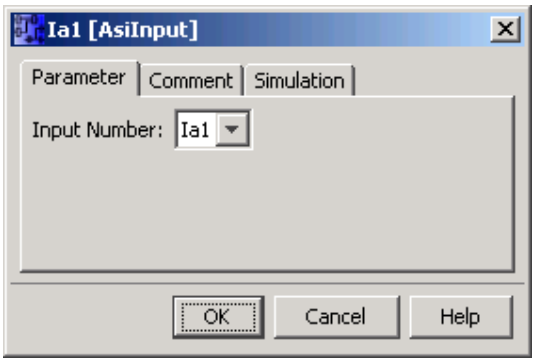
AS-Interface outputs

Outputs for the AS-Interface bus can be identified by the letter Qa (only available for the relevant LOGO! versions). Modular LOGO! devices as of the series 0BA3 do not distinguish between normal inputs and AS-Interface inputs. The user determines the type of the input, based on the inserted modules and the order in which they are installed. This is why AS-Interface outputs are only designated by the letter Q.

1.7.2 AS-Interface inputs



LOGO! versions of the type LB11 can be connected directly to an AS-Interface bus.



AS-Interface inputs are named Ia. The block number of an AS-Interface input is determined by the hardware structure.

Circuit programs with AS-Interface I/Os are converted for use in the modular LOGO!. Information is found in the section "Converting circuit programs".

Overview

1.7.3 AS-Interface outputs

LOGO! versions of the type LB11 can be connected directly to an AS-Interface bus.

AS-Interface outputs are named Qa. The block number of an AS-Interface output is determined by the hardware structure.

The output always carries the signal of the previous program cycle. This value does not change within the current program cycle.

Circuit programs with AS-Interface I/Os are converted for use in the modular LOGO!. Information is found in the section "Converting circuit programs".

Overview

2 User interface

2.1 User interface - Overview

User interface and programming interface

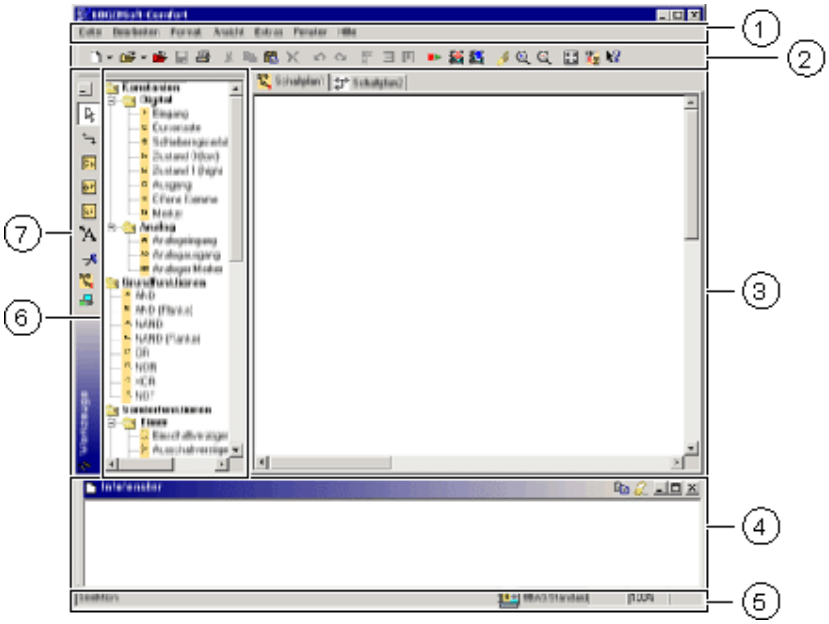
LOGO!Soft Comfort V6.0 starts with the empty user interface of LOGO!Soft Comfort.

Click on this icon: 

Result: LOGO!Soft Comfort creates a new, empty circuit program.

You now see the complete user interface of LOGO!Soft Comfort. The programming interface for creating your circuit programs occupies the greater part of the screen. The icons and logical links of the circuit program are arranged on this programming interface.

To help you to maintain an overview of large circuit programs, the right side and the bottom of the programming interface contain scroll bars, which you can use for vertical and horizontal scrolling of the circuit program.



1. Menu bar
2. Standard toolbar
3. Programming interface
4. Info box
5. Status bar
6.
   - Constants and connectors
   - Basic functions (only FBD Editor)
   - Special functions

⑦ Programming toolbar

Menu bar

The menu bar is located at the top of the LOGO!Soft Comfort window. Here, you can find various commands for editing and managing your circuit programs, as well as functions for defining your default settings and for transferring of the circuit program to and from LOGO!.

Toolbars

LOGO!Soft Comfort provides the following three toolbars:

- Standard toolbar
- Programming toolbar
- Simulation toolbar

Standard toolbar

The standard toolbar is located above the programming interface. After its start, LOGO!Soft Comfort shows you a reduced standard toolbar that provides only the essential functions.

The standard toolbar provides direct access to the essential functions of LOGO!Soft Comfort.

After you have opened a circuit program for editing on the programming interface, you can see the complete standard toolbar.

You can use the icons to create a new circuit program or to download, save and print out an existing program, cut/copy and paste objects, or initiate data transfer to and from LOGO! devices.

You can use the mouse to select and move the standard toolbar. The toolbar is always snapped onto the top of the menu bar when you close it.

Programming toolbar

The programming toolbar is located at the left of the screen. Its icons can be used to change to other editing modes, or for quick and easy creation or editing of a circuit program.

 or 

You can drag and drop the programming toolbar to another location with the mouse. The toolbar is always snapped onto the top of the menu bar when you close it.

The LAD Editor no longer contains the Basic function (SF) icon, because you create logical "AND" and "OR" links by interconnecting individual blocks.

Simulation toolbar

This toolbar is only relevant for the simulation of circuit programs. Further information is found here.

Info box

The Info Window, located at the bottom of the programming interface, displays information and notes, as well as the LOGO! devices recommended by the Tools -> Determine LOGO! function for use in your circuit program.

Status bar

The status bar is located at the bottom of the program window. It shows the currently active tool, the program status, the zoom factor, the page number of the circuit diagram and the selected LOGO! device.

2.2 Description of the Info Window

Content

The Info Window shows:

- Error messages generated at the start of simulation
- LOGO! devices determined by the Tools -> Determine LOGO! menu command or the function key [F2]
- The date and time of the message
- The name of the circuit program for which the message was generated

If you have opened more than one circuit program, you can determine to which program the message belongs.

At the start of simulation mode, the function analyzes the circuit program with regard to its resources and the LOGO! to be used. The resources used and the errors that occurred are displayed in the Info Window.

The Info Window displays all information in successive order. Use the scroll bar to browse all the information pages. All information is deleted from the Info Window when you close LOGO!Soft Comfort.

Operation

You can open and close the Info Window with the View -> Info window menu command or the [F4] function key. The Info window is usually positioned at the bottom of the programming interface. You can move it with the mouse and snap it onto the top of the programming interface, in the same way as you move the toolbars. You can move the window by drag and drop, or move it out of LOGO!Soft Comfort to open it as a separate window.

A quick way of increasing/reducing the size of the Info Window

Editing the texts in the Info window

You can delete selected messages from the Info window or copy them to other applications. You can also write personal comments in the Info Window.

Use the mouse to select a text from the Info Window and use this icon to copy it to the clipboard of your operating system.

This icon can be used to delete the content of the Info Window.

How to use the Info Window texts for your documentation

2.3 Description of the status bar

The status bar is split into five sections and contains useful information about your circuit program.

1. Information field. Displays the currently used tool, for example.
2. Displays your selected LOGO! by means of a LOGO!Soft Comfort tooltip. If you have not yet selected a LOGO!, or want to change the selection, you can double-click on the LOGO! icon to call the Tools -> Select Hardware dialog.
3. Shows you the currently set zoom factor.
4. This last field displays the current circuit program page.

2.4 Function keys and shortcuts

We have implemented a number of function keys and shortcuts for frequently called functions, in order to support your work with LOGO!Soft Comfort.

Function keys in LOGO!Soft Comfort:

[F1] → Calls the context sensitive Online Help
[F2] → Tools -> Determine LOGO!
[F3] → Simulation start/exit
[F4] → View -> Info Window open/close
[F5] → Connector tool
[F6] → Constants and terminals tool
[F7] → Basic functions tool
[F8] → Special functions tool
[F9] → Text tool
[F10] → Opens the menu bar
[F11] → Cut/Join tool
[F12] → Simulation tool

Shortcuts in LOGO!Soft Comfort:

In the File menu:

[Ctrl+N] → File -> New (opens the default editor specified under Tools/Options/Editor)
[Ctrl+O] → File -> Open
[Ctrl+S] → File -> Save
[Ctrl+F1] → File -> Print preview
[Ctrl+P] → File -> Print
[Ctrl+-] → File -> Compare
[Alt+F4] → File -> Exit

In the Edit menu:

[Ctrl+Z] → Edit -> Undo
[Ctrl+Y] → Edit -> Redo
[Ctrl+X] → Edit -> Cut
[Ctrl+C] → Edit -> Copy

[Ctrl+V] → Edit -> Paste

[Ctrl+A] → Edit -> Select all

[Ctrl+G] → Edit -> Go to block

In the View Menu

[Ctrl+M] Select Connections

[Ctrl+mouse wheel] → View -> Zoom in

→ View -> Zoom out

In the Tools menu:

[Ctrl+D] → Tools -> Transfer: PC -> LOGO!

[Ctrl+U] → Tools -> Transfer: LOGO! -> PC

[Ctrl+H] → Tools -> Select Hardware

How to access functions via the shortcut menu

2.5 Toolbars

2.5.1 Standard toolbar - overview

The icons of the standard toolbar provide quick access to commands that are also available on the menu.

The following commands are found in the standard toolbar:

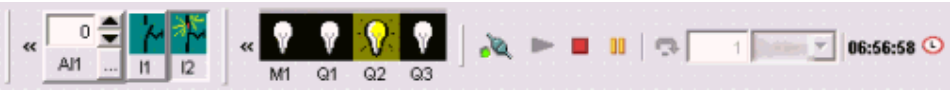
	File:	New
		Open
		Close
		Save
		Print
	Edit:	Cut
		Copy
		Paste
		Delete
		Undo
		Redo
	Format	Automatic
		Vertical
		Horizontal
	Tools:	Switch LOGO! Mode
		PC -> LOGO! (Download)
		LOGO! -> PC (Upload)
	View:	Select Lines
		Zoom in
		Zoom out
	File:	Properties, Page Layout tab
		Convert (LAD > FBD) Convert (FBD > LAD)
	Help:	Context-sensitive help

2.5.2 Simulation toolbar and status window

The toolbar

A toolbar pops up when you open the simulation mode. It contains the following icons:

- Icons (for example, switches) for operator control of the inputs
- An icon for the simulation of a power failure, for testing the switching response with reference to retentivity characteristics after power failure
- Icons (for example, bulbs) for monitoring outputs
- Simulation control icons
- Time control icons



Click << to hide a partial area of the toolbar. To show this area again, click >>.

Arranging the toolbar

You can move this I/O toolbar to the left, right, top or bottom of the programming interface by drag and drop, same as the other toolbars. If your program is exceptionally large and contains many I/Os, you can also drag and drop the I/O icons out of LOGO!Soft Comfort individually to open them in a separate window. This ensures a clear layout for your simulation.

Simulation control icons

	Start simulation
	Stop simulation
	Hold simulation (pause)

Time control

If you have programmed a time-sensitive circuit, you can use the time control to monitor the reaction of your circuit program.

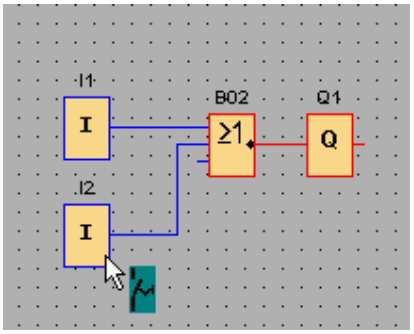
	Start simulation for a specific time or number of cycles. Set the period and the number of cycles using the following icons.
	Set the period and the time base for a time-limited simulation or set a specific number of cycles
	Display the current time in LOGO!Soft Comfort
	Modify the current time in LOGO!Soft Comfort

Status display

Prerequisite: The display of signal states and process variables is enabled under **Tools → Options: Simulation**.

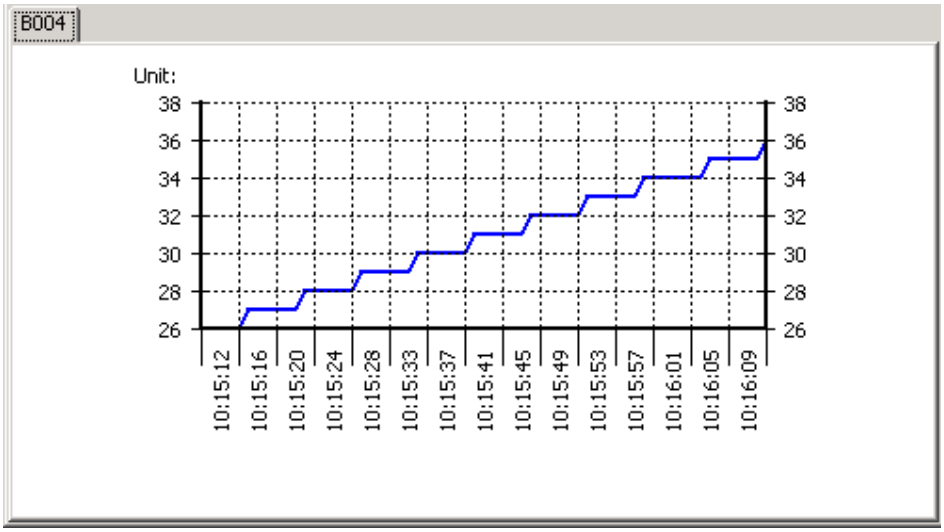
The colored indication lets you identify the "1" or "0" status of a connecting line. Default color of connecting lines carrying a "1" signal is red. Default color of connecting lines carrying a "0" signal is blue.

Example for the FBD Editor:



PI controller trend view

If your circuit program includes one or more PI controller function blocks, the simulation displays a trend view window of the analog output that shows the change in the value over time. The trend view is continually updated while the simulation is in progress. You can also display the process variable PV from the trend view window.



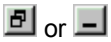
If your circuit program has more than one PI controller function block, the trend view has a separate tab for each PI controller analog output.

2.5.3 Programming toolbox

2.5.3.1 The programming toolbar - Overview

The programming toolbar contains integral icons for creating, editing and testing programs. Each one of these tools represents a programming mode, in which mouse operations have different effects.

The editing tools are not available as menu commands.

 Catalog of the elements of a circuit program open / close

 Selection tool

 Text tool

 Cut / Join

 Connector tool

 Constants and Terminals

  FBD Editor only:

Basic functions

 Special functions

 Simulation

 Online test

2.5.3.2 Catalog of circuit program elements

This catalog provides a hierarchical listing of all elements you can use to create your circuit program.

You can open and close this catalog via the icons  or  of the programming toolbar.

The handling of the catalog is self-explanatory.

 A quick and easy way of selecting blocks and placing these into your circuit program

2.5.3.3 Selection tool

You can use the selection tool to select and move blocks, text and connecting lines. You can select objects individually with a left-click; you can select multiple objects with [Ctrl]+Click, or you can use the mouse as a "lasso" to surround objects with a rectangle and capture them as a selection.

You can call the selection tool in any other tool by pressing the [ESC] key or by clicking on the icon in the programming toolbar.

Selecting objects

2.5.3.4 Text tool

This tool is used to insert or edit user-defined text objects in the programming interface. Instead of user-specific or block-independent text objects, you can also create labels which are assigned directly to specific blocks and are moved or deleted along with the relevant block. To create an associated label, click directly on the required block when the text tool is selected.

There can only be one associated label for each block. You can specify the font type, font size and font color for each individual label.

The width of the text box is the size of the longest line of text in the text box. To resize a text object or label, insert or delete carriage returns in the text string to make the text box the width and height that you require.

Documentation of the circuit program

2.5.3.5 Cut/Join

This tool is used to cut and join connections between blocks. To cut a connection, left-click toselect the relevant line while the Cut/Join tool is active. The connection is replaced at the blocks by a reference to the partner block. The reference is labeled with the page number, block number and the I/O of the partner block.

Cut connections

2.5.3.6 Connector tool

This tool is used to connect blocks. To do so, move the mouse pointer to a block input or output and press the left mouse button. Keep the mouse button pressed, drag the mouse pointer from your selected source terminal to the target terminal. Now release the mouse button to anchor the connecting line to both terminals. While the connecting line is being drawn, it is shown as a straight line between the first terminal and the mouse pointer. Once it is anchored, it appears as a combination of horizontal and vertical lines, which can be manipulated using the selection tool.

Connecting blocks

2.6 Menu bar

2.6.1 Menu bar - Overview

The menu bar commands contain administrative and editing functions for the circuit program of your LOGO! and context-sensitive help.

- File menu
- Edit menu
- Format menu
- View menu
- Tools menu
- Window menu
- Help menu

2.6.2 File menu

2.6.2.1 File menu - Overview

The **File** menu command contains commands for file management. Included are also commands for downloading, saving or creating circuit programs, for setting general file properties and for printing.

- New
- Open
- Close
- Close All
- Save
- Save As
- Page Setup
- Print Preview
- Print
- Properties
- Compare
- Exit

Available in the LAD Editor only:
Convert (LAD > FBD)
Available in the FBD Editor only:
Convert (FBD > LAD)

2.6.2.2 File -> New

 → Menu command **New**

The command opens a new window with an empty programming interface for programming in LAD or FBD, depending on your set mode. Depending on your default setting, a window opens with a number of tabs in which you can specify the properties of the circuit program you are going to create. This window can also be called later to either enter or modify the properties with the File -> Properties menu.

Program sections that have previously been placed on the clipboard by means of the cut or copy functions remain on the clipboard and can be pasted into the new circuit program.

An icon for this menu command is also found in the standard toolbar.

Switching between LAD and FBD

You select the editor for creating a new circuit program from the Tools -> Options: Standard Editor menu command.

Select either "FBD Diagram Editor" or "LAD Editor" from the Standard Editor drop-down list. New circuit programs will be created in either LAD or FBD according to your selection.

2.6.2.3 File -> Open

→ File **Open**

The command opens a dialog box from which you can select and open a previously created circuit program for further editing on the programming interface. Circuit programs created in LOGO!Soft Comfort have the file extension *.lsc. The loaded circuit program is opened in a new window.

An icon for this menu command is also available in the standard toolbar.

Circuit programs of LOGO!Soft Standard

The user can also import files created with LOGO!Soft Standard (file name extension = *.lgo). Use the "File type" menu item to select the type of file you want to display. LOGO!Soft Comfort will generate a graphical layout of the circuit program.

Alternatives

You can also open a circuit program by one of the following means:

- In Windows you can also drag and drop a LOGO!Soft Comfort circuit program file to the programming interface. When you "release" this file on the programming interface, LOGO!Soft Comfort opens it in a new window.
- If you double-click a file with the extension *.lsc or *.lld in the file manager, LOGO!Soft Comfort automatically opens the file.

What happens with the clipboard content?

Program objects previously copied to the clipboard with the cut or copy functions are stored in the clipboard and can be pasted into the new circuit program.

Recently-opened files

At end of the **File** menu you see a list of files that were recently opened in LOGO!Soft Comfort.

2.6.2.4 File -> Close

→ File **Close**

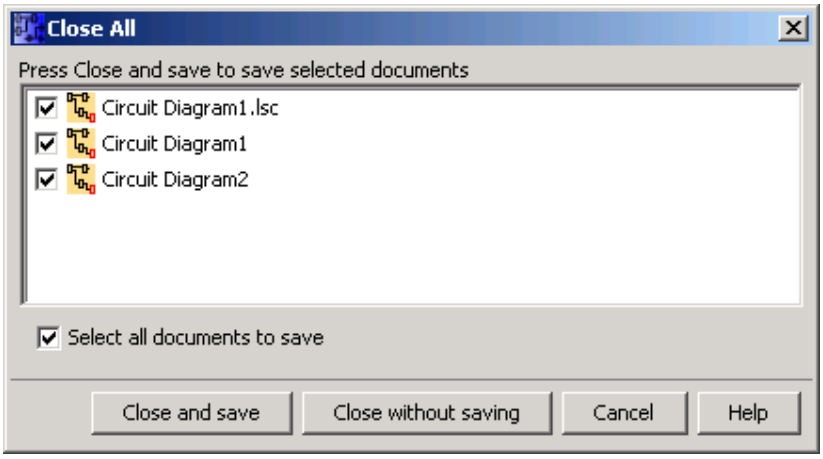
Click on the **Close** menu command to close the active window. If you have not yet saved the current circuit program, you are prompted to do so.

The standard toolbar also contains an icon for this menu command.

As an alternative, you can right-click on the tab of a circuit program and select the **Close** menu command from the shortcut menu.

2.6.2.5 File -> Close all

A click on the **Close all** menu command closes all open windows. If you have not yet saved one or more of the current circuit programs, you are prompted in a dialog to select the program to be saved. All selected programs will be saved.



2.6.2.6 File -> Save

 → File **Save**

When you initially save a newly created program, a window opens in which you can specify the path and file name under which you want to save your circuit program. Details are found under File -> Save as.

If you are saving a modified version of an existing program, a Quick Save is performed. The old version of the circuit program is overwritten by the revised version: the new program is saved to the same path and name as the source file.

The standard toolbar also contains an icon for this menu command.

As an alternative, you can right-click on the tab of a circuit program and select the **Save** menu command from the shortcut menu.

2.6.2.7 File -> Save As

A dialog box opens for you to specify the path and file name under which the current circuit program is to be saved. This allows you to save modified programs under a different name or folder, and thus keep previous versions for retrieval.

The types of files that you can save are listed below:

- LOGO!Soft Comfort file FBD(*.lsc)
- LOGO!Soft Comfort file LAD (*.lld)
- LOGO!Soft Standard file (*.lgo)
- Portable Document Format (*.pdf)
- JPG file (*.jpg)
- Bitmap file (*.bmp)

The default LOGO!Soft Comfort file name extension for FBD programs is *.lsc or *.lld for LAD programs. You can also export your program to an older LOGO!Soft version if you select the file type *.lgo, the program format of LOGO!Soft Standard. Additional information such as graphic information on block positions is not taken into account here. If your circuit program contains functions not supported in LOGO!Soft Standard, LOGO!Soft Comfort generates an export error message.

You can also save the circuit program in a graphical format, e.g. *.jpg, *.bmp or *.pdf, the AcrobatReader document format for program documentation and presentation. However, note that such files do not contain logic elements; you cannot reopen these in LOGO!Soft Comfort.

The AcrobatReader format offers a special feature. Saving your circuit program in *.pdf format gives you an AcrobatReader document that is absolutely identical to the hardcopy of your program. For example, you could distribute this document to users who do not have LOGO!Soft Comfort, and thus enable them to view your project in AcrobatReader and make hardcopies.

As an alternative, you can right-click on the tab of a circuit program and select the **Save as** menu command from the shortcut menu.

2.6.2.8 File -> Page Setup

This command opens a dialog box in which you can specify the page settings for creating circuit programs. Here you can specify the paper format, page margins or whether to print in portrait or landscape format.

LOGO!Soft Comfort offers multi-page printout feature, with the position of page breaks indicated on-screen. The print area is user-definable.

You can paginate your circuit program via the File -> Properties menu.

The settings made at this point have no effect on the printer settings. Select the printer setup command via the File -> Print menu. Finally, you can specify the scope of your hardcopies with Tools -> Options: Print.

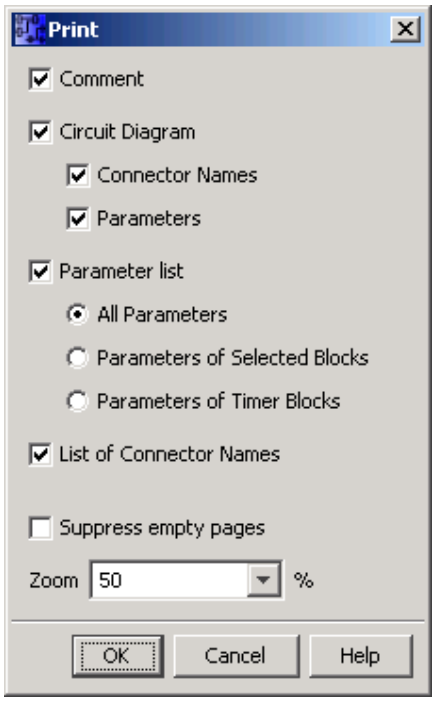
2.6.2.9 File -> Print preview

The print preview option shows what a hardcopy of your circuit program is going to look like. Choose the relevant icons to scroll the pages, zoom the window or to start printing directly.

2.6.2.10 File -> Print

→ Menu command **Print**

This command opens a dialog from which you can select the scope of information to print. You can also open this dialog from the Tools -> Options: Print menu.



You can choose whether to print **comments** you entered under File -> Properties comment or not.

You can also choose to include or omit **connection names** and parameters.

If you require a **parameter list**, you can also choose whether to include the parameters of all blocks, all selected blocks or only the special timer block.

Finally, you can print out a **list of connection names**.

The **Suppress empty pages** option allows you to exclude pages that do not contain any graphical objects.

Because blank pages are included in the page numbering, gaps will appear if these are not printed.

In the printer dialog, you can specify your default printer and the print properties. In the control panel of your computer you can specify advanced printer settings.

The AcrobatReader *.pdf format provides a further print option. You can save your program in AcrobatReader document format and distribute it to users who do not have LOGO!Soft Comfort and can then use AcrobatReader to view and print out your circuit program.

The standard toolbar also contains an icon for this menu command.

Set the page format under File -> Page Setup.

2.6.2.11 File -> Properties

The properties dialog contains the following tabs:

- General
- Comment
- Statistics
- Page Layout
- Parameter

2.6.2.12 File -> Properties: General

In the **General** tab of the Properties dialog, you can enter details of the current circuit program. There are input boxes for project-related and internal company data. You can quickly and easily specify the version of your circuit programs with the help of this dialog.

As an alternative to a text string for your company name, you can specify a *.bmp or *.jpg file with your company logo for the Company field. This feature lets you create a customized layout of your program files.

By setting the **Show at new file** check box, you are shown a flag that indicates where to input the specifications described above each time you create a new circuit program.

How to identify your circuit program version

2.6.2.13 File -> Properties: Comment

In the **Comment** tab, you can enter a description of the circuit program or notes relating to it. When you print your circuit program using the File -> Print menu command, you can choose to print this comment on a separate printed page.

2.6.2.14 File -> Properties: Statistics

The **Statistics** tab shows the creation date of the circuit program and the last author.

2.6.2.15 File -> Properties: Page Layout

→ Menu command **Page format**

In the **Page Layout** tab, you can specify how and on how many pages to print your circuit program. You can preview the pagination in this tab. If you choose more than one program page, the page breaks are indicated by white lines on the programming interface. Your circuit program is later printed out according to this pagination. Please note that connections extending to other pages are simply cut off when you print the hardcopy. We recommend you create cross-references by splitting the file at this position with the help of the Cut/Join tool. In the File -> Page Setup menu you can specify the paper size, page alignment and margins.

The standard toolbar also contains an icon for this menu command.

2.6.2.16 File -> Properties: Parameter

When the circuit program is transferred, all **Parameter** tab specifications are also transferred to LOGO! and then saved there.

In the **Program name** field, a program name with up to 16 characters can be entered for the circuit program. After transfer, the circuit program will then be shown on the LOGO! display under this name.

You can assign a password to your circuit program or change or delete an already assigned password. In order to assign a new password, you must enter the password in the two text boxes for **New password** and then confirm with OK. The password can have a maximum of 10 characters. To change the password, you must first enter the existing password in the **Old password** text box and the new password in the two boxes for **New password** and then confirm with OK. You can delete your assigned password at any time. To do this, enter the existing password in the box **Old password** and leave the two **New password** boxes empty. Then confirm with OK.

The password protects your circuit program on LOGO!. You can open or edit circuit programs from LOGO!Soft Comfort at any time regardless of whether the program is password-protected or not. For password-protected circuit programs, you must enter the password to view or change the program on LOGO!, or to load the circuit program from LOGO! to the computer.

With **Display content on LOGO! after power on** you can set what is shown on the LOGO! display when you switch LOGO on. Your choices are as follows:

- Display the date and time.
- Display inputs and outputs.

With **Behavior of analog outputs in STOP mode** you can set what the analog outputs of LOGO! should issue when LOGO! is in STOP mode. Your choices are as follows:

- AQ1 and AQ2 retain their last values.
- AQ1 and AQ2 are set to values that you define.

With **Set analog outputs value range type** you can set the range for analog outputs. Your choices are as follows:

- 0-20 mA / 0-10V
- 4-20 mA

With **Set AI3 and AI4 position** you can choose to enable the optional AI3 and AI4 on the LOGO! 0BA6 devices that have an option of four analog inputs. For LOGO! 0BA6 devices

that support four analog inputs, you can choose how many to use. Your choices are as follows:

- 2 AIs: Only AI1 and AI2 corresponding to input terminals I7 and I8 are available for use in your circuit program.
- 4 AIs: AI1 and AI2 corresponding to input terminals I7 and I8 are available for use in your circuit program. Additionally, AI3 and AI4 corresponding to input terminals I1 and I2 are available for use.

The presence of tabs and fields on the Properties dialog depend on the LOGO! module that you select from the Tools->Select Hardware menu command.

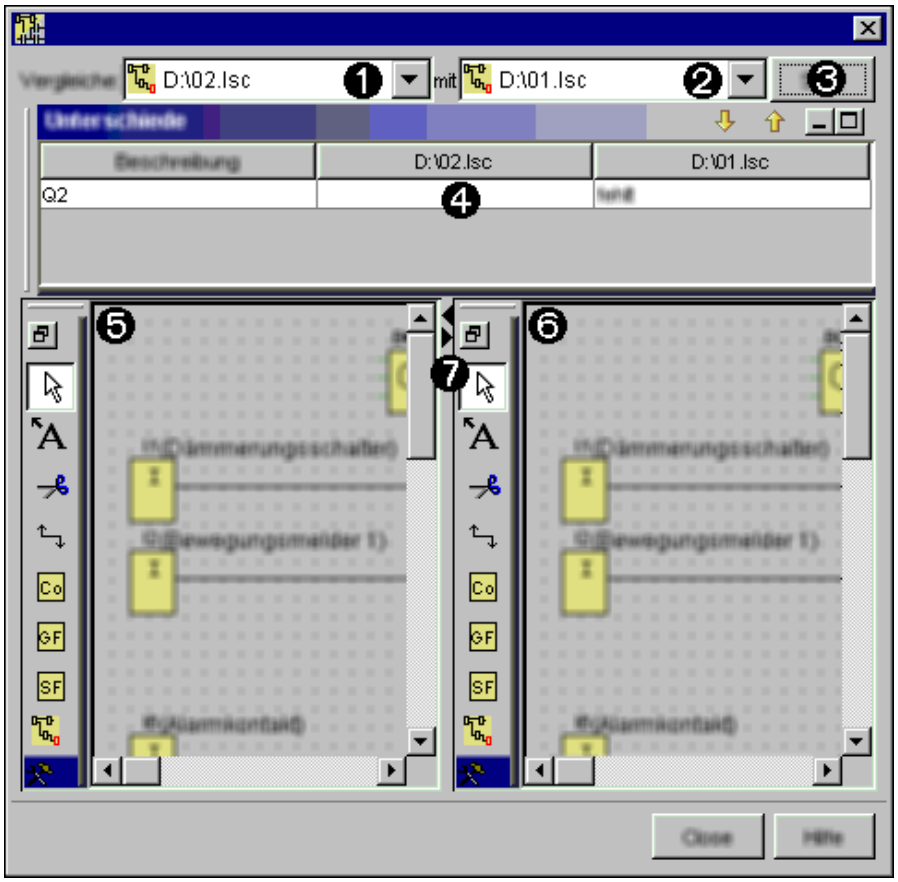
2.6.2.17 File -> Compare circuit programs

This function can be used to compare two circuit programs.

LOGO!Soft Comfort does not recognize differences in the graphical block layout and in comments.

Prerequisite

- The circuit programs must be open in LOGO!Soft Comfort. As an alternative, you can also compare a program opened in LOGO!Soft Comfort with a program on the LOGO! device.
- You can compare only circuit programs of the same type, e.g.: *.lsc with *.lsc.



Comparison

Procedure:

Step	Legend	Action
1	①	Select the first circuit program.
	⑤	Display of the first circuit program
2	②	Select the second circuit program you want to compare with the first one.
	⑥	Display of the second circuit program
	⑦	The two arrow icons can be used to increase the display area of one circuit program. However, the display area for the other circuit program is thus reduced.
3	③	Click on the **Start** button
4	④	The table indicates the following differences if found in the two circuit programs: • Number of blocks • Block parameters • Additional/missing blocks • Connections • Different hardware

Editing

All options are available to you for separate editing of the two circuit programs.

Different hardware

A considerable number of messages may be output if you have configured different LOGO! devices in the circuit programs you want to compare.

In this case, match the LOGO! devices in your programs.

Note

When you compare circuit programs with internal markers, in some circumstances LOGO!Soft Comfort reports more differences than there really are.

A quick and easy way of zooming your circuit program window

How to access functions via the shortcut menu

2.6.2.18 File -> Convert (LAD > FBD)

Use this function to convert your circuit diagram from LAD to FBD.

The following rules apply to the conversion from LAD to FBD:

- A series circuit of contacts is converted into an AND block.
- A parallel circuit of contacts is converted into an OR block.
- User-defined comments are not included, as their position in the circuit diagram can not be defined based on blocks.
- Crosslinks (connections where a block output is connected to multiple block inputs and at least one of the inputs is connected to multiple block outputs) are converted into an OR block.
  Inputs for the OR block are all block outputs of the crosslink.
  The output of the OR block is connected to all block inputs of the crosslink.
- Internal flags are resolved, and the current paths are linked.

Overview: File -> Convert (FBD > LAD)

2.6.2.19 File -> Convert (FBD > LAD)

Use this function to convert your circuit diagram from FBD to LAD.

The following rules apply to the FBD to LAD conversion:

- An AND block is converted into a series contact circuit
- An OR block is converted into a parallel contact circuit
- Comments for basic functions are not applied in LAD, as a basic function is converted into multiple contacts. Thus, the comment cannot be assigned definitely.
- In LAD, input comments are assigned to all contacts of this input.
- User-defined comments are not included, as their position in the circuit diagram can not be defined based on blocks.
- XOR blocks must be converted into corresponding LAD logic consisting of positive and negative contacts.

Note

When converting, the total number of blocks in your circuit program can sometimes increase. This could cause the permitted number of blocks for your LOGO! to be exceeded.

It is therefore not always possible to convert from FBD to LAD.

Remedy: Under **Tools > Select Hardware**, select the hardware series **0BA6**. Start converting to LAD. Then under **Tools > Determine LOGO!**, see which hardware series is compatible with the circuit program.

Overview: File -> Convert (FBD > LAD)

2.6.2.20 File -> Message Text Settings

Use this command to configure message text settings that apply to all message texts. The global message text settings are defined as follows:

- **Use new feature:** LOGO!Soft Comfort V6.0 provides many new features for message texts. If you want to use these new features, select this check box. If you want to use message texts with the functionality prior to V6.0, do not select this check box. Note that once you select new features, you cannot revert back to the previous functionality without loss of your message text data.
- **Character Set 1:** This specifies the primary character set for message texts. Message texts can be composed from characters from either Character Set 1 or Character Set 2. Either character set can be the default for configuring message texts.
- **Character Set 2:** This specifies the secondary character set for message texts. Message texts can be composed from characters from either Character Set 1 or Character Set 2. Either character set can be the default for configuring message texts.
- **Analog input filter timer:** This time determines the frequency at which LOGO! refreshes the analog values in a message text. The value is in milliseconds.
- **Ticker timer:** For messages that tick, the ticker timer determines how fast or slow a message text ticks or scrolls on the LOGO! Display or LOGO! TD. The value is in milliseconds, and you can use the adjacent button to access a scroll bar to set the ticker timer.

Supported Character Sets

LOGO!Soft Comfort, the LOGO! Display, and the LOGO! TD support the following character sets:

Character Set in LOGO!	Common Name	Supports Languages	Internet Reference
ISO-8859-1	Latin-1	English German Italian Spanish (partly) Dutch (partly)	http://en.wikipedia.org/wiki/ISO/IEC_8859-1
ISO-8859-5	Cyrillic	Russian	http://en.wikipedia.org/wiki/ISO/IEC_8859-5
ISO-8859-9	Latin-5	Turkish	http://en.wikipedia.org/wiki/ISO/IEC_8859-9
ISO-8859-16	Latin-10	French	http://en.wikipedia.org/wiki/ISO/IEC_8859-16
GB-2312 / GBK	Chinese	Chinese	http://en.wikipedia.org/wiki/GB2312

See the message text function description for details on how these settings apply.

2.6.2.21 File -> Exit

LOGO!Soft Comfort will be closed.

If you are actually editing a circuit program or havenyet saved it, a window opens upon exit.

In this window you can state which circuit programs are to be saved. Alternatively, you can exit LOGO!Soft Comfort without saving circuit programs. To do this, click **Close without saving**.

A quick way of closing LOGO!Soft Comfort without saving the data

2.6.3 Edit menu

2.6.3.1 Edit menu - Overview

In the Edit menu you will find commands for editing your circuit program. Basic commands for the creation and editing of a circuit program are included in the icons of the programming toolbar.

- Undo
- Redo
- Delete
- Cut
- Copy
- Paste
- Select All
- Goto block
- Bring to front
- Send to back
- Input/Output names
- Block properties
- Block properties (all blocks)
- Cut connections

2.6.3.2 Edit -> Undo

 → Edit **Undo**

This command allows you to undo actions carried out on the programming interface. Position your mouse pointer on the undo menu command and hold it there briefly. The tooltip opens and shows you the actions you can undo by clicking on the menu item. Currently you can undo up to 30 actions.

The standard toolbar also contains an icon for this menu command.

2.6.3.3 Edit -> Redo

 → Edit **Redo**

The redo reverts the last undo action. Click on the menu command to view the tooltip for the action to be redone.

The standard toolbar also contains an icon for this menu command.

2.6.3.4 Edit -> Delete

 → Edit **Delete**

The command deletes selected objects, without copying them to the clipboard. You can retrieve deleted objects with the **Undo** function.

2.6.3.5 Edit -> Cut

 → Edit **Cut**

The command deletes one or more selected objects, for example, blocks and/or connecting lines, from the programming interface and copies them to the clipboard.

The standard toolbar also contains an icon for this menu command.

2.6.3.6 Edit -> Copy

 → Edit **Copy**

The command is used to copy one or more selected objects, for example, blocks, text, or connecting lines, to the clipboard.

The standard toolbar also contains an icon for this menu command.

2.6.3.7 Edit -> Paste

The command copies the clipboard content to the programming interface. The insert position is either below the previously selected object or a position determined with a mouse click.

The standard toolbar also contains an icon for this menu command.

 → Edit Paste

You can only paste the clipboard content if sufficient resources are available. Blocks require a certain amount of resources, depending on the block type. An error message is generated if your system does not provide sufficient resources.

Connecting lines with open ends can not be pasted. They can only be pasted if they connect two blocks and were copied to the clipboard together with those blocks.

2.6.3.8 Edit -> Select all

The command is used to select all objects (blocks, connecting lines and labels) on the programming interface.

2.6.3.9 Edit -> Go to block

You can use this command to view a list of all blocks used, including information about the block numer, block name and block type. You can enter a short description in the **Block number** line (for example, I1, I2, B004, etc.) If the block number that you entered is in the circuit program, LOGO!Soft Comfort hightlights the specified block in the list and on the programming interface.

You can also select a block from the list directly to highlight it on the programming interface.

2.6.3.10 Edit -> Bring to front

You can use this command to bring one of a number of overlapping objects to the foreground.

2.6.3.11 Edit -> Send to back

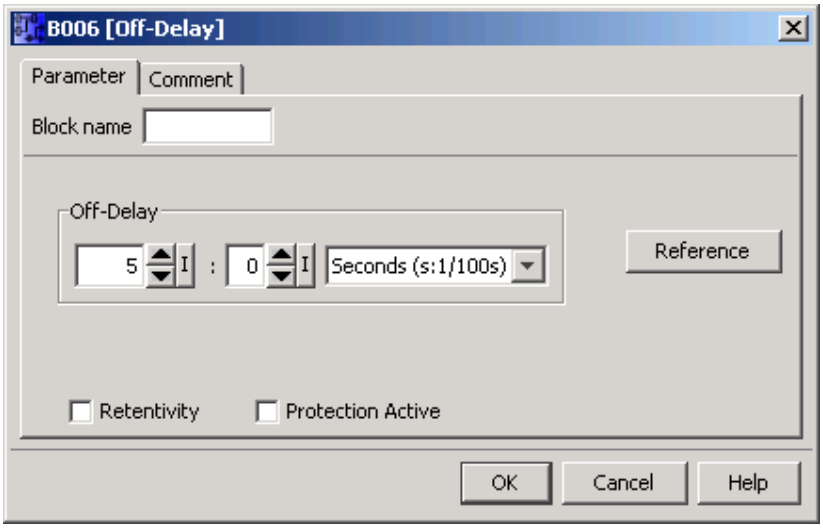
You can use this command to bring one of a number of overlapping objects to the background.

2.6.3.12 Edit -> Input/Output Names

This command enables you to enter names for the input and output terminals (connector names). Call the Tools -> Options -> Screen menu to specify whether to display the connector names on the programming interface. Call the Tools -> Options -> Print command to open a dialog to specify whether to include the connector names and connection list in the printed copy of your circuit program.

2.6.3.13 Edit -> Block properties

This command displays the block properties of a block that is selected on the programming interface.



The block properties dialog box consists of several tabs. Every block has a Comment tab, where you can enter relevant block comments. For some blocks a Parameter tab is available where you can describe specific block parameters. Input simulation parameters are configured from the Simulation tab.

You can also call the block properties by right-clicking on the required block, and selecting the block properties menu command from the shortcut menu.

2.6.3.14 Edit -> Block properties (all blocks)

This command opens a window with two sections. In the left section, you can see all the blocks that are used in your program. Click on a block to view its corresponding parameters in the right column. You can now edit these parameters and set the changes by clicking the **Apply** button.

The modified block is displayed in blue in the selection list, if you do not accept the parameter changes made and select another block. All changes are discarded by clicking on the **Cancel** button. You confirm your entries and exit the dialog with **OK**.

If you select the "Select block from drawing" check box, then when you select subsequent blocks from the list they are highlighted in the circuit diagram.

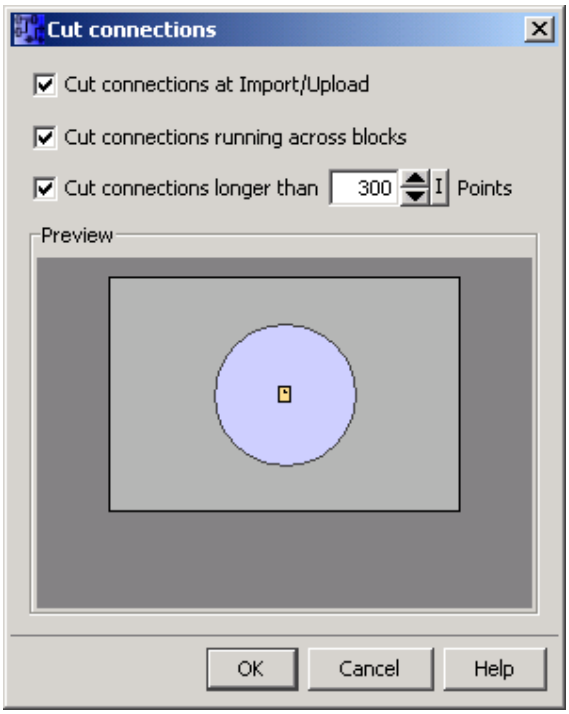
Special functions

Basic functions

Constants and Terminals

2.6.3.15 Edit -> Cut connections

You can choose to cut connections via the Cut / Join tool either manually or automatically.



In this dialog you can specify one or both of the following types of connections to be cut:

- Connections routed across a block
- Connections exceeding a configurable length

Confirm these settings with **OK,** and LOGO!Soft Comfort will cut connections accordingly.

If you select the **Cut connections during import/upload** check box, the settings described above will also be applied in the following cases:

- When you upload (transfer) a circuit program from LOGO! to LOGO!Soft Comfort
- When you import (open) a circuit program created with LOGO!Soft Standard or LOGO!Soft Comfort

2.6.4 Format menu

2.6.4.1 Format menu - Overview

This menu provides formatting options for labels and function groups. You can define the font, the font size and style as well as the alignment of selected objects.

- Font
- Align
- Format Grid
- Snap to Grid

2.6.4.2 Format -> Font

Here you can specify the font type, size, style, and the text color. To redefine the format of existing text objects, you first have to select them. You can then specify the font attributes to suit requirements and click on OK to apply the new formats.

You can choose to set particular default formats. New text objects are then displayed with these default settings. To return to the standard settings, click the **Standard** button.

2.6.4.3 Format -> Align

This command offers various options of aligning labels and function blocks:

Vertical

Horizontal

Automatic

2.6.4.4 Format -> Align -> Align Vertical

 → Format **Align vertical**

Selected objects are aligned vertically to the objects with the highest block number or to the last selected object you have placed into the circuit program.

The standard toolbar also contains an icon for this menu command.

2.6.4.5 Format -> Align -> Align Horizontal

 → Format **Align horizontal**

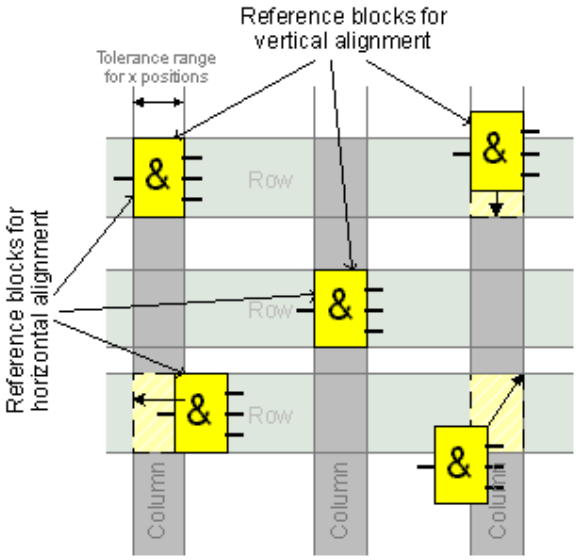
Selected objects are aligned horizontally to objects with the highest block number or to the last object placed into the circuit program.

The standard toolbar also contains an icon for this menu command.

2.6.4.6 Format -> Align -> Align Automatically

→ Menu Format **Align automatically**

Selected objects are automatically aligned in vertical and horizontal direction. Slightly offset or adjoining blocks are aligned along a common line. Reference for vertical alignment is the relevant upper block of a column. Reference for horizontal alignment is the block at the extreme left of a line.



The standard toolbar also contains an icon for this menu command.

2.6.4.7 Format -> Format grid

This tool helps you to organize the various objects of your circuit program on the programming interface. The grid is switched on by default.

You can adjust the grid pattern (spacing) in increments of 5 points.

If you have enabled the "Snap Block to Grid" function, LOGO!Soft Comfort aligns the objects with the relevant grid intersections. This helps you to avoid a vertical or horizontal offset of the objects. To finely position objects on your programming interface, disable the "Snap Block to Grid" function.

If you have enabled the "Snap Connection to Grid" function, LOGO!Soft Comfort aligns the connection lines between objects with the relevant grid intersections. To allow connection lines to not align with grid lines on the programming interface, disable the "Snap Connection to Grid" function.

You can hide the grid with the Visibility check box.

You can select the "Use as Default" check box to keep your settings as the default for LOGO!Soft Comfort circuit programs.

2.6.4.8 Format -> Snap to grid

If you have made changes to the grid pattern, or inserted objects while the grid was disabled, the position of objects may be offset when they are aligned to the grid points. Call this command to correct the offset of selected objects and to realign them.

2.6.5 View menu

2.6.5.1 View menu - Overview

From the View menu, you can set the zoom factor for the display of your circuit, and decide to show or hide various display windows.

- Zoom
- Zoom In
- Zoom Out
- Toolbars
- Select Lines
- Info Window
- Status bar
- Tooltips

2.6.5.2 View -> Zoom

LOGO!Soft Comfort offers a variety of options for enlarging or reducing the size of the circuit program display. By selecting Zoom, you open a dialog box in which you can set the zoom factor from a default list or in the relevant box.

If you choose an unfavorable zoom factor, the objects may appear out of focus on your screen. You should therefore use the default zoom factors wherever possible. This effect has no influence on the layout of the printed circuit program.

A quick and easy way of zooming your circuit program window

2.6.5.3 View -> Zoom In

 → View **Zoom In**

The zoom factor is increased by defined increments:

25 (min) → 50 → 75 → **100** (default) → 150 → 200 → 250 → 300 → 400 (max)

The standard toolbar also contains an icon for this menu command.

A quick and easy way of zooming your circuit program window

2.6.5.4 View -> Zoom Out

 → View **Zoom Out**

The zoom factor is reduced in defined increments:

400 (Max) → 300 → 250 → 200 → 150 → **100** (default) → 75 → 50 → 25 (Min)

The standard toolbar also contains an icon for this menu command.

A quick and easy way of zooming your circuit program window

2.6.5.5 View -> Toolbars

This command lets you hide or show selected toolbars.

- Standard: Hide/show the standard toolbar
- Tools: Hide/show the programming toolbar

2.6.5.6 View - > Select Lines

 → View **Select Lines**

With this setting all connections (lines) that lead to or away from a selected block are shown in color.

If you select a single connection with this setting, then the selected connection is highlighted in color.

Under **Tools > Options > Screen** you can set whether the connections should also be labeled. Under **Tools > Options > Colors** you can set the colors to use to display the connections.

Tools -> Options: Screen

Tools -> Options: Colors

2.6.5.7 View -> Info Window

This menu command can be used to show or hide the Info Window. You can also use the function key [F4].

2.6.5.8 View -> Status bar

This menu command can be used to hide or show the status bar.

2.6.5.9 View -> Tooltips

In LOGO!Soft Comfort, you can use the mouse-over-button function to display the icon name, which represents the tooltip.

This helps you to quickly recall the function of the icon, without having to call the menu or the help.

2.6.6 Tools menu

2.6.6.1 Tools menu - Overview

The options menu provides the following menu commands:

- Transfer
- Determine LOGO!
- Select Hardware
- Simulation
- Simulation parameters
- Online test
- Connect modem
- Disconnect modem
- Options

2.6.6.2 Tools -> Transfer

Prerequisite for data transfer

The interface used to link LOGO! to the PC must be configured via the Tools -> Options: Interface menu.

- Otherwise LOGO!Soft Comfort will return an appropriate error message. Click on the **Select New interface** button from any of the Transfer menu command dialogs to open the options dialog for the configuration of the communication interface. If the wrong interface or no interface is set, you can determine the PC interface to which LOGO! is connected or you can start a search for the interface.
- For further information on how to connect the LOGO! to your PC with a USB interface, refer to the LOGO! manual.
- The LOGO! must be connected to the PC with the PC cable or be accessible by modem.
- LOGO! may neither be in RUN nor in editing mode.

0BA0-0BA3:

The LOGO! must be prepared for data transfer with the PC/Card -> PC <-> LOGO! setting. For further information, refer to the LOGO! manual.

Menu commands

The **Tools** → **Transfer** menu contains the following menu commands:

Tools -> Transfer: PC -> LOGO!

Tools -> Transfer: LOGO! -> PC

Tools -> Transfer: Switch LOGO! Mode

Tools -> Transfer: Set clock

Tools -> Transfer: Summer/Winter time

Tools -> Transfer: Hours counter

2.6.6.3 Tools -> Transfer -> PC -> LOGO!

 → Tools -> Transfer: **PC -> LOGO!**

This command is used to download a circuit program created on the PC in LOGO!Soft Comfort to the LOGO! module. The name of the program transferred to LOGO! is specified in the File -> Properties menu. LOGO!Soft Comfort uses the interface you specified in Tools -> Options: Interface to transfer the circuit program.

The standard toolbar also contains an icon for this menu command.

Preparations

Prior to the download, the system determines the minimum LOGO! version required for your circuit program. The modular LOGO! always provides all available I/O resources for your circuit program. It is up to the user to install an appropriate number of expansion modules in the base device.

Error messages

If the circuit program cannot be downloaded to the available LOGO!, the transfer is aborted and an error message is displayed. The user is informed of unknown LOGO! versions and then has the choice of continuing or canceling the download. A message in the status bar reports the successful download.

Transfer messages are displayed in the status bar and in the Info Window.

Detailed error messages are displayed in the Info Window.

Password

If you have assigned a password to your circuit program, then you will be asked to enter this password before you transfer to LOGO!. The circuit program will then only be transferred to LOGO! if you have entered the correct password.

2.6.6.4 Tools -> Transfer -> LOGO! -> PC

 → Tools -> Transfer: **LOGO! -> PC**

The circuit program is imported from LOGO! to LOGO!Soft Comfort. LOGO!Soft Comfort uses the interface you specified in Tools -> Options: Interface to transfer the circuit program. Transfer messages are displayed in the status bar and in the Info Window.

The standard toolbar also contains an icon for this menu command.

Missing graphical information

A program imported from LOGO! to LOGO!Soft Comfort does not contain any graphical information for the block layout on the programming interface. A suitable layout for the circuit program is therefore generated automatically. The generated circuit diagram corresponds with the layout in the LOGO!Soft block diagram, except that multiple instances of the same block are not displayed, but are instead identified by means of the block connectors.

The blocks are always arranged at the top left corner of the programming interface. If necessary, you can use the scroll bars to bring the circuit program into view.

Cutting connections

If you have set the "Cut connections during import/upload" check box under Tools -> Options: Cut connections, the relevant connections are cut during the upload from LOGO! to the PC, according to the rules set in this dialog.

Password

At the start of the upload of a password protected circuit program from the LOGO! to the PC, the user is prompted to enter the password. If an invalid password or no password is entered, the transfer is aborted with an error message.

2.6.6.5 Tools -> Transfer -> Switch LOGO! Mode

 → Tools -> Transfer -> **Switch LOGO! Mode**

This special function is only available with devices of the 0BA4 hardware series or later.

When you click on this symbol you change the mode of a connected LOGO! from STOP mode to RUN mode or from RUN mode to STOP mode.

2.6.6.6 Tools -> Transfer -> Set Clock

This menu option can be used to view and set the date and time of the connected LOGO!.

Click on **current time** to apply the system time of the PC in LOGO!Soft Comfort.

Manual input of values

You can enter the values directly from the keyboard by clicking on the number input box instead of clicking on the arrow icons of the date and time setting function. LOGO!Soft Comfort automatically corrects any invalid date values.

2.6.6.7 Tools -> Transfer -> Summer time/Winter time

This menu command lets you set an automatic conversion of the summer and winter time for the LOGO! clock.

When you enable summer/winter time conversion, you can specify a country-specific time conversion:

- EU: European Union
- UK: United Kingdom of Great Britain and Northern Ireland
- US1 / US2: United States of America
- Australia
- Tasmania
- New Zealand
- Freely adjustable: customized switchover dates and times

For the “Freely adjustable” selection, you specify the month and day of the switchover. The start time of summer time is 02:00 + the entered time difference; the end time is 03:00 – the entered time difference.

The United States of America redefined the Daylight Saving Time / Standard Time calendar in 2007. US1 is the convention in effect prior to 2007 and US2 is the convention defined in 2007 where Daylight Saving Time is in effect from 2:00 a.m. on the second Sunday in March until 2:00 a.m. on the first Sunday in November according to the local time zone.

Note that this function is only supported in LOGO! devices as of the series 0BA3.
The US2 selection is only supported in LOGO! devices as of the series 0BA6.

2.6.6.8 Tools -> Transfer -> Hours Counter

Use this menu command to read the hours counter of LOGO!.

You can also fetch the hours counter from LOGO! devices with a password protected program without having to enter a password.

Only the LOGO! devices >= version 0BA3 support this function. Also, you can only transfer the hours counter of a LOGO! that is not equipped with a red module, because you delete the LOGO! program if you remove this module.

2.6.6.9 Tools -> Transfer -> Clear User Program and Password

Use this command to clear the circuit program in the LOGO! device and the program password if a password exists.

You must respond to a confirmation dialog to ensure that you do intend to clear both the circuit program and the password (if configured) from the LOGO! device. After you confirm the command, LOGO!Soft Comfort performs the clear operation. If you do not confirm the command, LOGO!Soft Comfort performs no action. The circuit program and password (if configured) remain in the LOGO! device.

LOGO! devices prior to version 0BA6 do not support this function. If you attempt this command on an earlier device, LOGO!Soft Comfort displays a message that states that the device does not support this function.

2.6.6.10 Tools -> Transfer -> Set LOGO! TD Power-on Screen

Use this command to configure a power-up screen for the LOGO! TD. From the dialog, you can also use the Read button to read a previously-configured power-up screen that is stored in the memory of the LOGO! Basic module, or you can use the Write button to write the currently-configured screen to the LOGO! Basic module memory. The LOGO! Basic module updates the LOGO! TD with the power-up screen that is stored in memory.

Use the keyboard to type characters into the display window. The power-up screen can only contain simple text strings.

LOGO! devices prior to version 0BA6 do not support this function. If you attempt this command on an earlier device, LOGO!Soft Comfort displays a message that states that the device does not support this function.

2.6.6.11 Tools -> Determine LOGO!

When you click on this menu command, LOGO!Soft Comfort calculates the minimum LOGO! version requirements for the LOGO! circuit program. The result is shown in the status bar. The Info window displays all versions to which you can download the program. You can also use the function key [F2] to execute this menu command.

2.6.6.12 Tools -> Select Hardware

You have two options of selecting the device when you create your program in LOGO!Soft Comfort:

- You can first create your program and then determine the required LOGO! by using the Tools -> Determine LOGO! dialog.
- You can first determine the LOGO! version for which you want to create your circuit program by calling the **Tools** → **Select Hardware** dialog. You can also call the device selection dialog by double-clicking on the LOGO! icon in the status bar.

The device selection dialog shows you which blocks and memory resources are available to you.

If you have already created a circuit program or are using some blocks, the hardware selection dialog offers you only the LOGO! devices you can operate with the currently used blocks.

2.6.6.13 Tools -> Simulation

Introduction

Program simulation allows you to test a program and modify its parameters. This ensures the download of a fully functioning and optimized program to your LOGO!.

Simulation mode

Click on the **Simulation** icon in the programming toolbar to start the simulation. This changes the program to simulation mode.

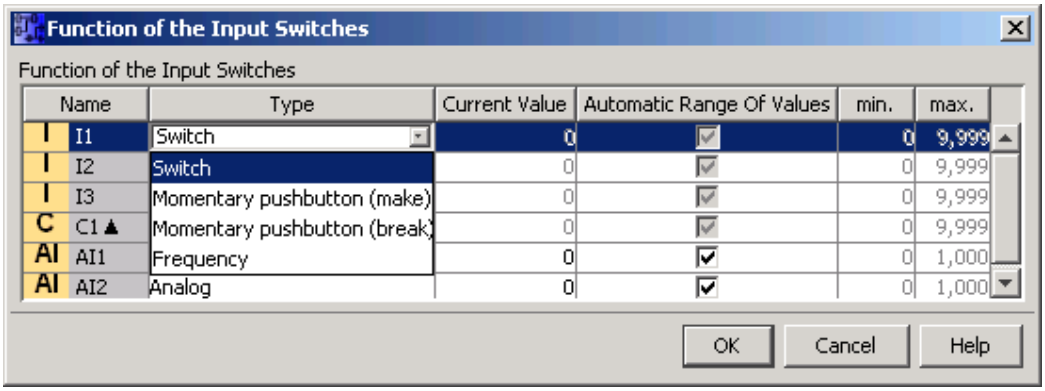
 → **Simulation**

The active icon is highlighted in simulation mode. By clicking once again on the **Simulation** icon, or on any other icon of the programming toolbar, you exit simulation mode and open another tool mode (for example, for selecting or inserting blocks).

At the start of simulation mode, the program is verified and the results are output to the Info Window.

2.6.6.14 Simulation Parameters: Input functions

Select the **Tools** → **Simulation Parameters** command to configure the response of an input. The dialog shows only the inputs actually used in your circuit diagram.



You have four options:

- Switch
- Momentary pushbutton
- Frequency (not for analog inputs)
- Analog (analog inputs only)

Switch

A switch latches when actuated and is released by actuating it once again.

Momentary pushbutton

A momentary pushbutton is only active while it is held down. The contact opens as soon as you release the button.

You can determine a make or break action for the pushbuttons.

Frequency (not for analog inputs)

The frequency of the frequency input can be preset or changed while simulation is running. The device frequency is expressed in Hz. A frequency input is a special case in this context, because it is only useful in conjunction with the threshold trigger SFB.

Analog (analog inputs only)

You can preset the value for the analog input or modify it while running in simulation mode. The unit of the analog value to be set corresponds with the default process variable. The range corresponds with the specified measurement range, provided the option "Auto range" is selected. The value range corresponds in this case with the measurement range of the function connected to the input. The analog input represents a special case, because it should only be used for analog SFBs.

Refer to the information in the "Analog value processing" section for help on analog block parameters.

Name column

Digital inputs are designated I.

Analog inputs are designated AI.

Settings

When you save your circuit program, the settings for circuit simulation are included. Thus, you do not need to enter the simulation parameters once again when you exit and reopen your circuit program.

While simulation mode is active, you can click on a digital input in the list of input switches to change its settings. From the drop-down list select Switch, Momentary pushbutton (make), Momentary pushbutton (break) or Frequency. Use the OK button to enter your settings.

While simulation mode is active, you can also right-click on a digital input in the circuit program and edit its block properties. In the Simulation tab of the block properties dialog, you can select the type of the input as described above. Use the OK or Apply button to enter your settings.

2.6.6.15 Tools -> Online Test

The online test and simulation modes allow you to monitor the execution of your circuit program and how it reacts to the various input states.

Difference from simulation mode

In simulation mode you execute your circuit program on the PC. To do so, you do not require a LOGO!. The status of inputs can be preset on the PC.

During an online test, the circuit program is executed on a LOGO!. The user monitors this "work" of the LOGO!. The status of the inputs corresponds with the actual states at the LOGO! inputs.

Prerequisite for an online test

Your PC must be linked to a LOGO!.

The circuit program to be tested can be in either FBD or LAD format and must be transferred to LOGO!.

The circuit programs in LOGO!Soft Comfort and on the LOGO! must be identical. Upload the program from the LOGO! to your PC, or download the program from your PC to the LOGO! if necessary.

You can monitor the parameters of up to 30 blocks. The number of blocks you can monitor simultaneously decreases when you monitor blocks that contain a high number of parameters (for example, analog SFBs).

To start the online test

1. Select the Tools → Online Test menu command
2. If the LOGO! is in STOP, start it with the Start  button
   Result: The LOGO! executes your circuit program.
3. Now start the monitoring mode. 
4. Select the blocks whose parameters you want to monitor.
   Result: You are shown "live" how the parameters of the selected blocks change.

Switching the LOGO! to STOP

If you want to stop the LOGO! via LOGO!Soft Comfort, click on the Stop icon  .

Devices of the OBA4 series and later support Online tests.

Possible errors

The following types of errors can occur:

- Your LOGO! does not support the online test.
  Remedy: Install a LOGO! device of the latest series.
- The programs on your PC and on the LOGO! are different.
  Remedy: Upload the circuit program from the LOGO! to your PC, or download the program from your PC to LOGO!.
- You are attempting to monitor too many parameters/blocks simultaneously.
  Remedy: Reduce the number of simultaneously monitored parameters/blocks.
- The communication between your PC and LOGO! goes down.
  Remedy: Re-establish the connection.

2.6.6.16 Tools -> Connect Modem

You can use modems to download and upload circuit programs between LOGO!Soft Comfort and LOGO! devices. LOGO!Soft Comfort supports 11-bit modems that use standard AT commands.

Select the **Tools -> Connect Modem** menu command to start the process of configuring modems to use between your PC with LOGO!Soft Comfort and a LOGO! Basic module. LOGO!Soft Comfort displays a Modem Information dialog that provides an overview of the configuration process.

You must complete a series of dialogs to configure your modems. After you configure the modems and make the connections, you can download and upload circuit programs between LOGO!Soft Comfort and LOGO! across a telephone connection between the modems.

The steps for connecting and configuring modems are listed below:

- Select Modem
- Select Remote Configuration
- Configure Remote Modem Command
- Configure Remote Modem
- Select Local Configuration
- Configure Local Modem Command
- Configure Telephone Number

From the modem configuration dialogs, you can use the Next button to proceed through the dialogs, or the Previous button to return to an earlier step.

2.6.6.17 Tools -> Connect Modem - Select Modem

From the Select modem dialog, you can select check boxes to configure the remote modem or the local modem. The remote modem is the one connected to the LOGO! Basic module and the local modem is the one connected to the computer with LOGO!Soft Comfort.

If you have previously configured the remote modem or local modem from LOGO!Soft Comfort, the dialog displays the names of the modem as configured from LOGO!Soft Comfort. You can click the Detail button to display specific information about the LOGO!Soft Comfort configuration of a modem.

If you have configured the remote modem by some means other than LOGO!Soft Comfort, you can disregard the displayed settings. LOGO!Soft Comfort will use the configuration in the actual modem. If you have not yet configured the remote modem, or choose to change the existing configuration, then select the Remote modem check box.

For the local modem, LOGO!Soft Comfort will use the settings as displayed by the Detail button, or you can select the Local modem checkbox to edit the configuration.

If you need to configure or change the configuration of both modems, then select both check boxes.

If you have already configured both modems but need to set the telephone number, do not select the check boxes. When you click the Next button you will advance to the telephone number configuration dialog. Otherwise, when you click the Next button you will proceed to configuration of the modems that you selected.

2.6.6.18 Tools -> Connect Modem - Select Remote Configuration

From the Select remote configuration dialog, you select whether to edit or delete an existing configuration, if one exists, or to create a new configuration.

Click the Next button to continue with modem configuration.

2.6.6.19 Tools -> Connect Modem - Configure Remote Modem Command

From the Configure remote modem command dialog, you provide the name of the remote modem. You can select the check box to accept the default settings for the modem commands, or you can choose not to use the defaults and to edit the individual commands. The “Additional” field provides the opportunity to enter other modem commands. Separate each one with a blank if you add additional commands.

The commands are all standard commands. Refer to the documentation provided with your modem for specific command syntax.

Click the Save Configuration button to save this configuration for the remote modem.

2.6.6.20 Tools -> Connect Modem - Configure Remote Modem

The Configure remote modem dialog is the last dialog for you to complete remote modem configuration. After you ensure that the remote modem is connected to the LOGO! Basic module, follow these steps:

1. Select the Done check box.
2. Select the communications port to use.
3. Click the Configure remote modem button.

LOGO!Soft Comfort will complete the modem configuration and display a status message.

2.6.6.21 Tools -> Connect Modem - Select Local Configuration

From the Select local configuration dialog, you select whether to edit or delete an existing configuration, if one exists, or to create a new configuration.

Click the Next button to continue with modem configuration.

2.6.6.22 Tools -> Connect Modem - Configure Local Modem Command

From the Configure local modem command dialog, you provide the name of the local modem. You can select the check box to accept the default settings for the modem commands, or you can choose not to use the defaults and to edit the individual commands. The “Additional” field provides the opportunity to enter other modem commands. Separate each one with a blank if you add additional commands.

The commands are all standard commands. Refer to the documentation provided with your modem for specific command syntax.

Click the Save Configuration button to save this configuration for the local modem.

2.6.6.23 Tools -> Connect Modem - Configure Telephone number

From this dialog you can add and delete telephone numbers for the modem connection.

Click the Add New button to provide a name, telephone number, and description for a telephone connection. To add multiple telephone numbers, click Add New and enter additional telephone numbers as needed. To remove a telephone number from the list, select it and click Remove.

From this dialog, you can dial one of the telephone numbers. To establish modem communication across one of the telephone connections, follow these steps:

1. Verify that the local modem is connected to your computer and that the remote modem is connected to LOGO!, and click the Done check box.
2. Select a telephone number from the list.
3. Click the Dial button.

LOGO!Soft Comfort will establish the modem connection and display a status message. If any error occurs, LOGO!Soft Comfort will display an error message. Check your connections and your configuration if you receive an error. Refer to the documentation provided with your modem for additional information.

2.6.6.24 Tools -> Disconnect Modem

Use this command to disconnect an existing modem connection.

To configure modems, or to connect across configured modems, select the Tools -> Connect Modem menu command.

2.6.6.25 Tools -> Options: General

Here you can select various options for LOGO!Soft Comfort:

- Standard Editor
- Language
- Document view
- Screen
- Print
- Cut connections
- Interface
- Simulation
- Colors
- Look & Feel

2.6.6.26 Tools -> Options - Standard Editor

Here you define the default editor; that is, the FBD or LAD editor.

2.6.6.27 Tools -> Options: Language

Here you set the dialog language for LOGO!Soft Comfort.

To set and apply a new language, you must close and restart LOGO!Soft Comfort.

2.6.6.28 Tools -> Options: Document view

Here you determine whether to display the circuit programs in LOGO!Soft Comfort from a dialog tab or in windows.

The advantage of the window view is that you can arrange several circuit programs next to each other for easy comparison.

In the tab view, you can right-click on the tab to open a shortcut menu with the following menu items:

- Close
- Save
- Save as

2.6.6.29 Tools -> Options: Screen

This is where you perform all the settings to do with screen display.

Here you determine what you see in your circuit program:

- Comments
- Connector Names
- Block Parameters

Other possible settings:

- Antialiasing. With Antialiasing corners and edges appear softened.
- If you have **View > Select lines** switched on, with **Label marked lines** you establish that marked lines are to be labeled as shown in the following example:

  B007 > B006/2 means: the connection runs from block 7 to block 6 at pin 2. Connections are not labeled if the target block to which the connection is running is located in the immediate vicinity.
- You determine whether LOGO!Soft Comfort should note the size and position of dialogs that have been opened once.
- You also determine whether LOGO!Soft Comfort should note the entire working environment (position of windows, opened circuit programs etc.).

2.6.6.30 Tools -> Options: Print

Here you determine what is to be the printed from the following choices:

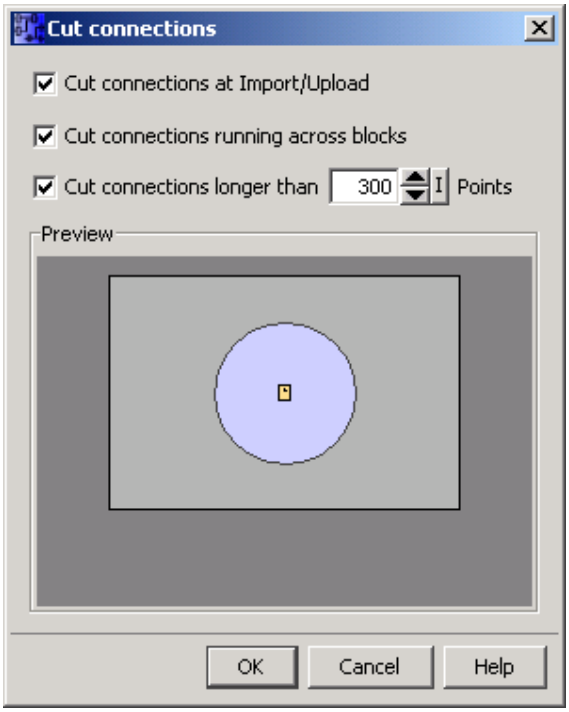
- The comment you have entered under File -> Properties: Comment
- Connector names and parameters
- A list of the parameters of all blocks, of all selected blocks, or only of the special timer functions
- List of connector names

Here you can also set whether empty pages should be suppressed and if the circuit program should be printed out enlarged or reduced.

This dialog is always displayed before you start printing.

2.6.6.31 Edit -> Cut connections

You can choose to cut connections via the Cut / Join tool either manually or automatically.



In this dialog you can specify one or both of the following types of connections to be cut:

- Connections routed across a block
- Connections exceeding a configurable length

Confirm these settings with **OK,** and LOGO!Soft Comfort will cut connections accordingly.

If you select the **Cut connections during import/upload** check box, the settings described above will also be applied in the following cases:

- When you upload (transfer) a circuit program from LOGO! to LOGO!Soft Comfort
- When you import (open) a circuit program created with LOGO!Soft Standard or LOGO!Soft Comfort

2.6.6.32 Tools -> Options: Interface

Specify an interface from the list, if you know exactly which one links the LOGO!.

If you are using a PC cable interface, but you do not know exactly to which interface the LOGO! is connected, you can let LOGO!Soft Comfort automatically detect the relevant interface.

2.6.6.33 Tools -> Options: Simulation

In simulation mode you can switch the display of signal states and process variables on or off. Switching it off improves the performance of your simulation because when you disable the display of signal states and process variables, there is no need for LOGO!Soft Comfort to calculate these values continuously.

2.6.6.34 Tools -> Options: Colors

Here you can define the color settings:

- Desktop color
- Colors of signal lines carrying a logical "1" or "0" signal in simulation mode
- Color of a selected line: You can set the color of a selected connection for each of the 4 maximum possible inputs and for the output of a block separately.

To restore the original settings, click the **Standard** button.

2.6.6.35 Tools -> Options: Look and Feel

Here you can personalize the layout of the LOGO!Soft Comfort user interface. Try it out!

How to display a corresponding tooltip for a function key

2.6.7 Window menu

2.6.7.1 Window menu - Overview

From the Window menu, you can arrange your circuit program windows on the desktop You can duplicate existing circuit programs and split the windows in order to obtain a clearer overview of large programs. The following window options are available:

- Arrange Vertical
- Arrange Horizontal
- Cascade
- Split Vertical
- Split Horizontal
- Undo Split

2.6.7.2 Window -> Arrange Vertical

You can tile several windows containing circuit program vertically on the programming interface.

This menu command is only available if you have set the window view instead of the dialog tab view via the Tools -> Options: Document view dialog.

2.6.7.3 Window -> Arrange Horizontal

You can tile several windows containing circuit programs horizontally on the programming interface.

This menu command is only available if you have set the window view instead of the dialog tab view via the Tools -> Options: Document view dialog.

2.6.7.4 Window -> Cascade

You can cascade several open windows containing circuit programs on your programming interface, starting on the upper left corner.

This menu command is only available if you have set the window view instead of the dialog tab view via the Tools -> Options: Document view dialog.

2.6.7.5 Window -> Split Vertical

If you have a large circuit program and want to view and compare widely distributed circuit objects, you can split the window vertically. The split, of course, affects only the window, but not your circuit program. You can use the scroll bars in the split windows to view or modify the various elements of your circuit diagram.

You can also split the window into several partitions if you consider it necessary. You can split each window several times, both in the horizontal and vertical direction; the partition in which the last mouse operation has taken place is the partition that is affected.

You can modify your circuit program in any area of the split window. These changes are, of course, executed throughout the circuit program, because only the window was split, and not the circuit program.

2.6.7.6 Window -> Split Horizontal

If you have a large circuit program and want to view and compare widely distributed circuit objects, you can split the window horizontally. The split, of course, affects only the window, but not your circuit program. You can use the scroll bars in the split windows to view or modify the various elements of your circuit diagram.

You can also split the window into several partitions, if you consider it necessary. You can split each window several times, both in the horizontal and vertical direction; the partition in which the last mouse operation has taken place is the partition that is affected.

You can modify your circuit program in any area of the split window. These changes are, of course, executed throughout the circuit program, because only the window was split, and not the circuit program.

2.6.7.7 Window -> Undo Split

You can use this menu command to undo all splits of a circuit program window.

2.6.7.8 Window -> Selection list

The selection list at the end of the Window menu shows you all the windows you have opened on the programming interface. You can use this selection list to quickly change between windows.

2.6.8 Help menu

2.6.8.1 Help menu - Overview

This menu provides you with help and information on LOGO!Soft Comfort.

- Content
- Context-sensitive help
- Update Center
- About

2.6.8.2 Help -> Contents

The Online Help

The Online Help quickly and reliably provides you with information about program configuration, tools and the creation of circuit programs with LOGO!Soft Comfort.

Topics of the Online Help

The user interface section describes the user interface with its toolbars and the LOGO!Soft Comfort menus in detail.

Refer to the tutorial for a quick and easy introduction to the basics of operating LOGO!Soft Comfort and its circuit programming features.

Towards the end of this section you will find an extensive practical example that takes you through all the steps of circuit program creation.

The sample applications section introduces a few applications for LOGO!.

The reference chapter contains the following subsections:

- The constants and terminals, basic functions (only FBD editor) and special functions subsections provide you with information about the various elements of a circuit program.
- The circuit programs subsection provides information on memory requirements, circuit program limits for LOGO!, and additional information about blocks.

In the Tips and Tricks section we have gathered information that supports your daily tasks with LOGO!Soft Comfort.

The Online Help naturally includes an **index** as well as a **full text search** feature for keyword and terminology based searches.

Help for blocks

If you double click on a block in the circuit diagram, you receive a window with parameters and settings for the block. If you then click on the Help button in this window you receive the Help for this block in its own window. This Help window is missing the following symbols: 

Remedy: Right-click on the block in the circuit diagram and select the **Help** menu command.

2.6.8.3 Help -> Context-sensitive help

 → Help **Context-sensitive help**

To call a help file on an object, first click on the context-sensitive help icon (see above) and then on the object.

Result: A window opens with information on this object.

You can also right-click on objects on the programming interface to call a corresponding help topic. The help entry in the shortcut menu called provides you with the required support.

The standard toolbar also contains an icon for this menu command.

2.6.8.4 Help -> Update Center

Update Center

The Update Center helps you to install additional languages, program add-ons, service packs and new versions for your LOGO!Soft Comfort.

Update and upgrade

If you update the software within the same main version, then this is an update. For example, LOGO!Soft Comfort Version 4.0 can be updated to Version 4.1. It is only possible to update via the Internet.

If you update the software to a higher main version, then this is an upgrade. For example, LOGO!Soft Comfort Version 5.0 can be upgraded to Version 6.0. You can upgrade from either the Update Center or a CD-ROM.

How to use the Update Center

Follow these steps to perform an update or upgrade:

1. Select whether you want to update LOGO!Soft Comfort via the Internet or using your local file system (CD-ROM, floppy or hard disk drive).
2. If you choose to update LOGO!Soft Comfort from your local file system, then you are prompted to enter the folder path in which the updates / upgrades are saved.
   If you update LOGO!Soft Comfort from the Internet, then the correct Internet address is already preset in the Settings Internet Update. If you are not connected directly to the Internet, then you must specify a proxy server. Consult your network administrator in this regard. An Internet connection is then created.
3. You are then shown all the updates / upgrades available for your software version. Select the desired updates / upgrades. If you are updating LOGO!Soft Comfort from the Internet, then your selected updates / upgrades will be downloaded and installed.
4. If you are updating LOGO!Soft Comfort via the Internet, you are prompted to manually close the Internet connection when you have completed these actions.

LOGO!Soft Comfort is closed automatically when the update / upgrade is completed. The functionality of the installed update / upgrade is available to you after you restart LOGO!Soft Comfort.

Possible errors

If, when installing an upgrade / update, you receive the error message **Does not agree with magic number**, this means the upgrade / update file **Setup.exe** has not been executed in full.

In this case, download the upgrade / update file from the Internet again and ensure the file is transferred in full.

2.6.8.5 Help -> About

The **General** tab displays the version number and the release version of your LOGO!Soft Comfort software.

The **System** tab provides you with information on the version of the Java Runtime environment used, the program paths, the installed operating system and on the memory used.

3 Tutorial

3.1 Prerequisites for working with the tutorial

We assume you are familiar with PC operation and that you know how to create a function block diagram. To download your circuit programs, you also need the PC cable for connecting the serial PC interface to your LOGO! device.

3.2 Getting started with program creation

3.2.1 Introducing the creation of circuit programs

You are going to learn the basics of working with LOGO!Soft Comfort, by creating a simple circuit program and simulating it on your PC. Towards the end of this chapter, you will find a few sample applications for LOGO!Soft Comfort, and information on how to prepare, transfer and archive your application.

If necessary, review the elements of the user interface before beginning the tutorial.

Factory door

Air-conditioning system

Heating control

Fill station

3.2.2 Creating a circuit program

3.2.2.1 Creating a circuit program

Creating programs with the help of the toolbars

In this intro section you require only the standard toolbar and programming toolbar.

To select a tool, drag the mouse pointer onto the icon and left click to select it. This selection is indicated.

Developing a circuit program

Follow these steps to develop a circuit program:

1. Create a new circuit program.
2. Select blocks
3. Place the blocks
4. Configure and comment the blocks
5. Connect the blocks
6. Optimize the circuit program
7. Save the circuit program

Please note that not all blocks are available under all circumstances.

3.2.2.2 Creating a new circuit program

You can start to create a new circuit program immediately after you have started LOGO!Soft Comfort.

To do so, click on the **File new** icon in the standard toolbar.

 → File new

LOGO!Soft Comfort then opens the FBD Editor (or the default editor specified under Tools/Options/Editor), and you can create the new circuit program in a new window on the programming interface.

Click on the small arrow on the right side of the **File new** icon to open the LAD or FBD Editor.

3.2.2.3 Selecting blocks

Your first step in programming a circuit diagram is to select the blocks for your circuit. Determine the order in which you want to insert the I/Os and the standard/SFB blocks.

Under Co in the programming toolbar, you will find the constants and terminals (only in the LAD editor), that is, a selection of I/Os and constant signals. Under BF, you will find the basic logic functions of Boolean algebra: standard digital logic blocks. Under SF you can find the special functions. You can also call the respective function groups via the function keys.

or [F6] → Constants / Terminals

or [F8] → SFBs

Only in the FBD editor:

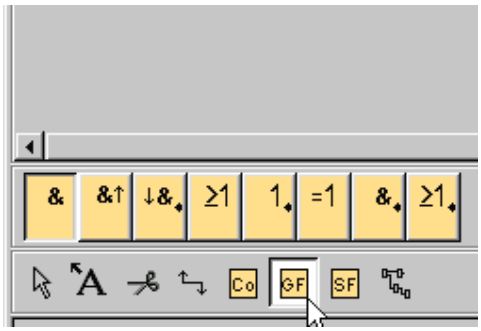
oder [F7] → Basic functions


A quick and easy way of selecting blocks and placing them into your circuit program

3.2.2.4 Placing blocks

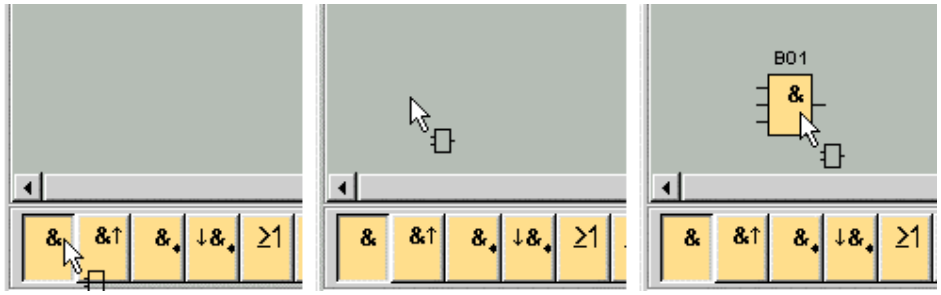
Click on the icon group that contains the required block or, as an alternative, press the function key. All blocks belonging to the selected function group are now shown below the programming interface.

Example for the FBD Editor:



You can insert the selected function on your programming interface with a simple mouse click. The first group function is set by default, and you can select other functions with the mouse before you place them.

Example for the FBD Editor:



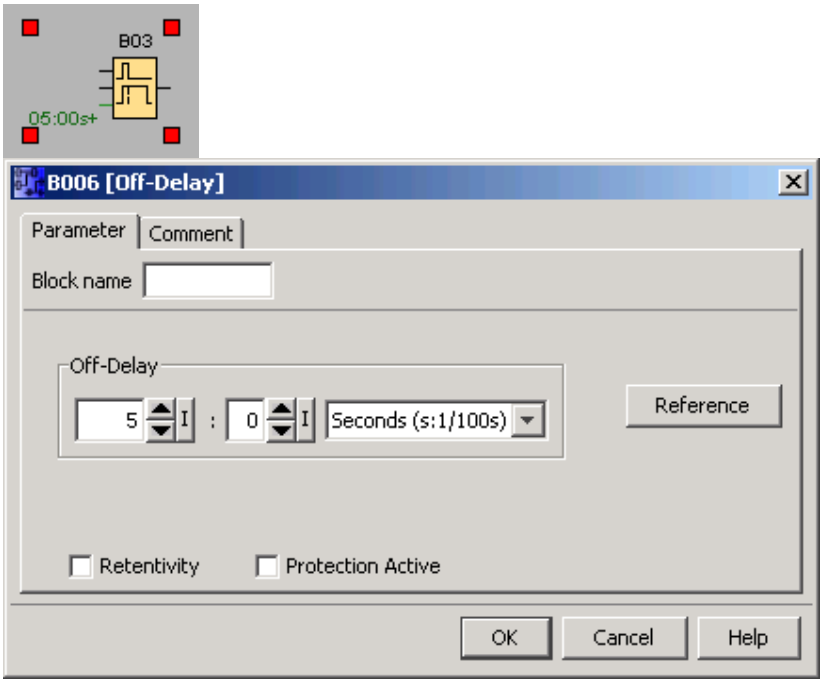
There is no need to align the blocks right away. A precise alignment of the blocks at this time does not make sense, unless you have interconnected them and entered the comments in your circuit program.

Information on block numbering is found here.

How to quickly and easily select blocks and place them into your circuit program

3.2.2.5 Editing blocks

Shortcut menus



A right-click on an object opens a shortcut menu that offers you various object editing options. The editing options depend on the selected object. Objects consist of not only blocks and connecting lines, but also the programming interface and toolbars.

You can also call help on the selected object in the shortcut menu.

Configuring blocks

Double-click with the left mouse button to configure block properties. The properties dialog includes a Comment tab as well as various parameter tabs for the SFBs, and for some of the basic functions and constants and connectors. You specify here the values and settings for your blocks. **Help** on the parameters of the relevant block can be called by clicking the help button.

Special functions can be recognized by the green lettering to the left of the block on the programming interface.

3.2.2.6 Connecting blocks

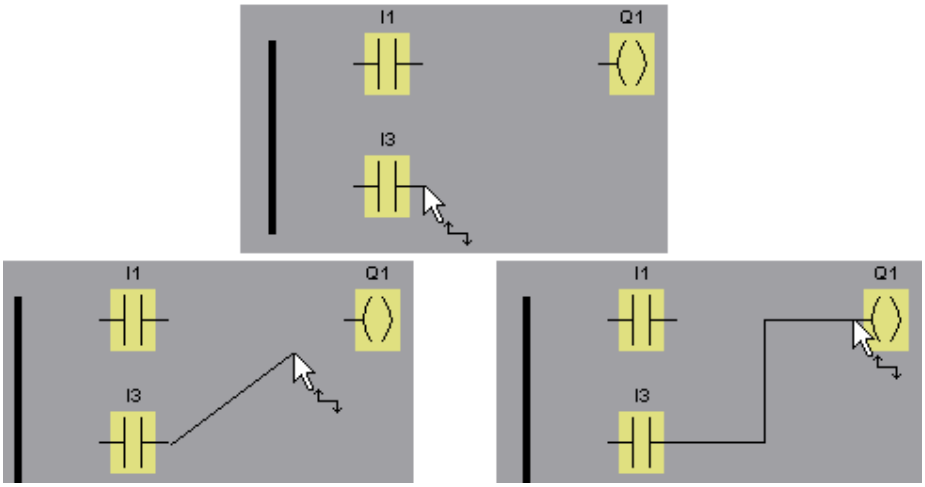
To complete the circuit diagram, you must interconnect the blocks. In the programming toolbar, select the block connection icon.

Example for FBD:

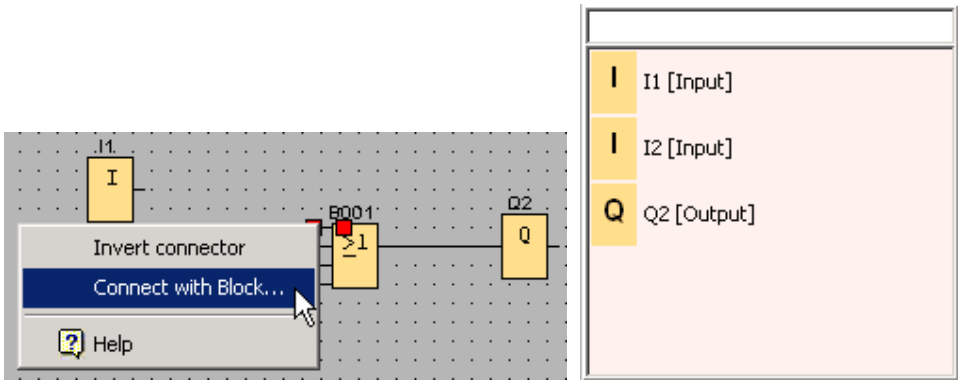
 → Connector

Position the mouse pointer on the block connector. Press the left mouse button and hold it. Move the pointer from the source connector to the target connector. Release the mouse button. LOGO!Soft Comfort connects the two terminals.

Example for LAD:



LOGO!Soft Comfort offers you a further option of connecting blocks when you right-click on the input or output of a block. In the shortcut menu, click the **Connect to block** menu command. This calls a selection list that contains all blocks available for your connection. Click on the relevant target block. LOGO!Soft Comfort then draws the connecting line. This method is especially useful for connecting a source to a target block over a greater distance on the programming interface.



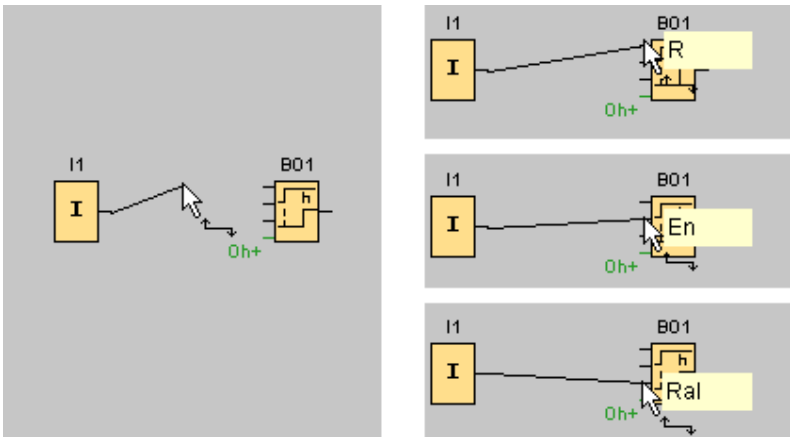
Note on the LAD Editor:

Do not forget to connect the I/Os to the bus bar on the left edge of the editor window.

Tips on connecting blocks

After the line is connected from an output to an input, or vice versa, a pop-up tool tip opens to show the connection. Release the mouse button to snap the line onto the indicated input.

You can refer to the short information (tool tips) in LOGO!Soft Comfort for additional help on circuit programming. Move the mouse pointer over a block and briefly hold it in this position. The name of the block is shown. The name of the block input appears when you move the mouse pointer onto the input.



To make it easier for you to interconnect blocks, a blue frame around the mouse pointer pops when it is "captured" by a pin.

Rules for connecting blocks

The following rules apply to the connection of blocks:

- You can connect a single input to multiple outputs.
- You cannot connect multiple inputs to a single output.
- You cannot interconnect I/O in the same path of a circuit program.Recursion is not permitted. Interconnect a flag or output if necessary.
- SFBs also have green "connectors". These do not represent connecting pins, but are used instead for assigning the parameter settings.
- Analog I/O cannot be connected to digital I/O.

Multiple connections

You can connect I/O to existing connections.

A quick and easy way of connecting blocks in large circuit programs

3.2.2.7 Availability of blocks

Hardware defaults

The memory space and the device series of your LOGO! determines:

- How many blocks you can use in the circuit program
- Which blocks you have available to create your circuit program

A LOGO! of the latest device generation is selected by default.

After you have created a circuit program, you can call the Tools -> Determine LOGO! menu command or press the function key [F2] to display an Info Window that shows you which LOGO! devices are available for executing your circuit program.

The blocks which are not available for your selected LOGO! are grayed out.

Optimizing the circuit program

Should you determine in the course of creating your circuit program that a LOGO! device is unable to handle your circuit program, you should first fall back on all the functional resources offered to you by the LOGO! device. You could, for instance, replace memory intensive blocks with a structure consisting of several blocks, which altogether require less memory space.

If the various optimization attempts are unsuccessful, then you can use an additional LOGO!, or optimize / simplify the functionality of your application.

3.2.3 Editing the layout

3.2.3.1 Editing and optimizing the layout

The circuit program is ready for use after you have inserted and connected the blocks. However, a slight touchup of your circuit is required in order to optimize your layout. You can reposition the inserted blocks and lines accordingly.

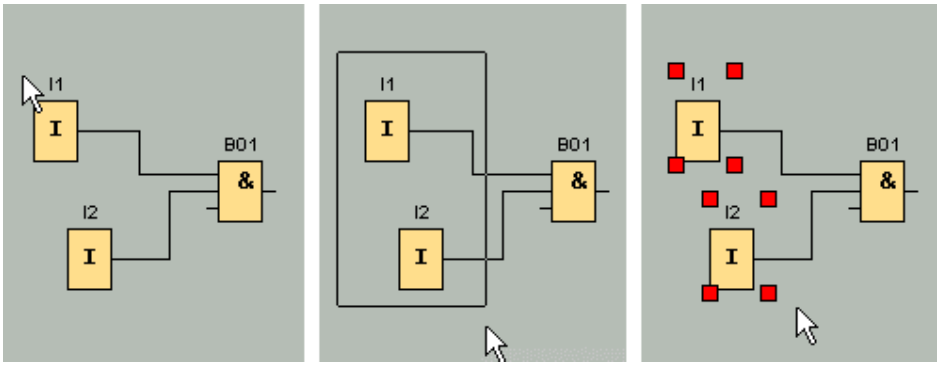
3.2.3.2 Selecting objects

Before you can move or align objects you must first select them. Click on the Selection Tool in the programming toolbar. You can also press the [ESC] key to activate the Selection.

 or **[ESC]** → Selection

Single blocks or connecting lines are selected simply by mouse click. Groups of blocks or connecting lines are selected by "capturing" them with the mouse pointer. To "capture" objects, keep the left mouse button pressed and draw a frame around them and then release the mouse button. The "captured" objects are highlighted by small red squares at the corners of the selected fields.

Sample for the FBD Editor:



In addition to the selection of single objects by a simple mouse click or highlighting object groups by "capturing", there is a further selection option: Under "optional selection", mark the objects one after the other by holding down the [Ctrl] key while you select the objects. To remove an object from the selection, hold down the [Ctrl] key and click on a selected object once again.

3.2.3.3 Editing selected objects

You can delete single or grouped objects with the [Del] key, or move them with drag and drop or use the keyboard. The cursor keys allow fine positioning in very small steps. However, the snap function must not be set in the Format -> Grid menu when doing so. You can also cut, copy and paste selected objects by means of the relevant toolbar icons.

 → **Cut a Selected Object**

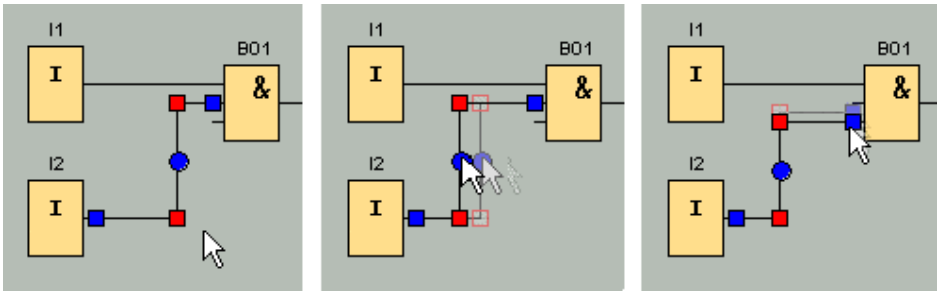
 → **Copy a Selected Object**

 → **Paste a Selected Object**

Editing selected connecting lines

A special option is offered for editing connecting lines. Selected connecting lines are indicated by round and square blue handles. The round handles can be used to move the lines at a right angle into the direction in which they extend. The square handles can be used to reassign the beginning or end of a line. The lines are moved by dragging the round handles.

Example for the FBD Editor:



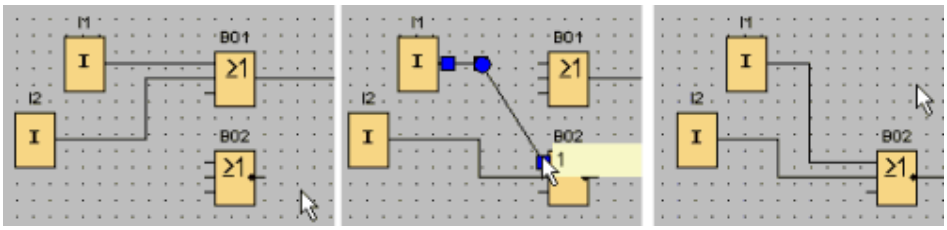
If the end of a connecting line is not assigned to a suitable target connector, it is automatically reconnected to its initial position after you "release" the mouse button.

3.2.3.4 Replacing blocks

How to replace a block in your circuit diagram with another function:

1. Insert the new block above or below the block you want to replace.
2. Rewire the connecting lines of the old block to the new one as described under Editing selected connecting linesedit_marked_objects.
3. After having rewired all the connecting lines, you can delete the old block and move the new block into this position.

Example for the FBD Editor:



By keeping to this block replacement order, you can maintain your connecting lines. If you first delete the old block, you also delete its connections, which means you have to recreate all connections.

3.2.3.5 Cutting connections

It may turn out to be difficult to interpret the layout of a large circuit, especially if it contains many line crossings. You can clean up your connection layout, using the "Cut/Join" tool of the programming toolbar.

 → **Cut/Join**

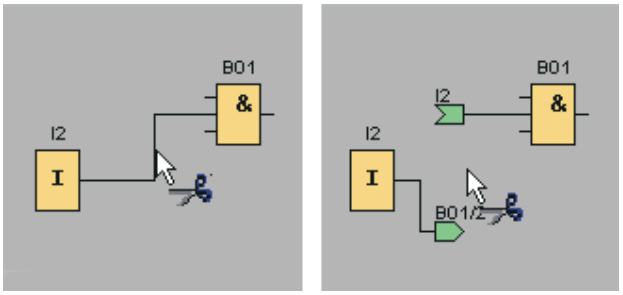
Click on a connection after you have called this tool. The selected connecting line is graphically split. However, the link between the blocks remains active.

The open ends of the cut connection are now shown with arrowhead icons, which indicate the direction of the signal flow. Above the icons, you can now see the cross-references, including the page number of the circuit diagram, the block name and the number of the block terminal that is connected to the open link.

Right-click on the line connecting the two blocks you want to cut, then select the cut command.

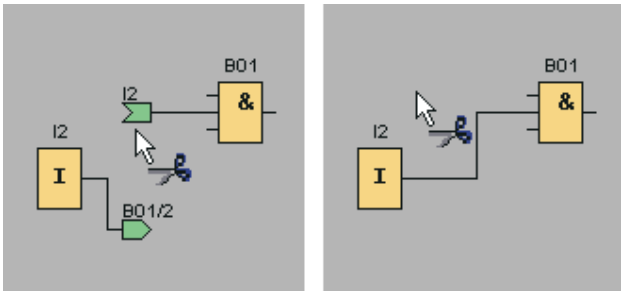
You can also cut a group of connections, using the Edit -> Cut Connections menu command. Before you cut any connections, you can also set the cutting criteria, for example, cut all connections routed through blocks.

Example for the FBD Editor:



The connection is closed again by clicking on its open end while the Cut/Join tool is active. Optionally, you can close the connection by right-clicking on an open end and calling the **Link** menu command.

Example for the FBD Editor:



You should not use this tool to edit smaller circuit diagrams. In most cases you can optimize the layout by repositioning the icons.

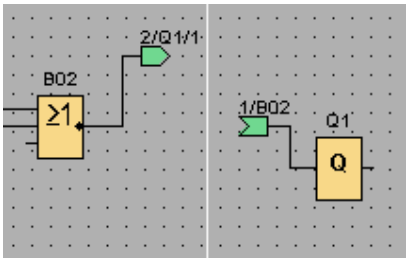
Applications and advantages

Large and complex circuit layouts may contain numerous line crossings, thus making it more or less difficult to interpret the circuit. In such cases, the Cut/Join button is a highly effective means of clearing up the circuit layout.

You can quickly jump to the partner connector by right-clicking the open end of a cut connection. This opens a shortcut menu, in which you can select the **GoTo Partner** menu command to jump to the partner end of the cut connection.

Another advantage of the Cut/Join tool is its utilization for circuits extending across more than one printable page, that is, across a page break. The connecting lines of two circuit blocks which are shown on different pages are cut without cross-reference. However, if you cut such connections using the Cut/Join tool you generate a cross-reference pointing to the source or the connection target.

Example for the FBD Editor:



3.2.4 Documentation and saving

3.2.4.1 Documentation of the circuit program

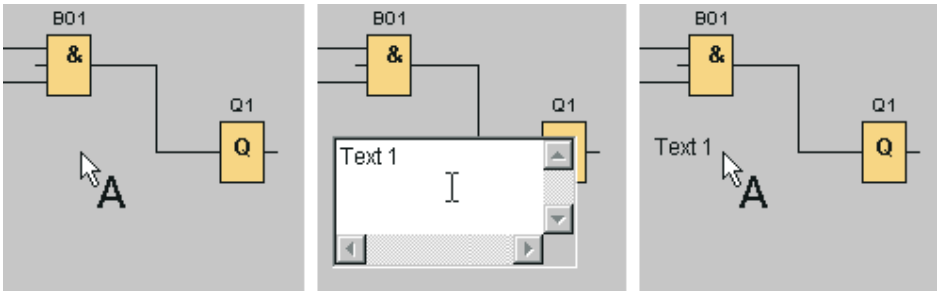
Labels

You can create block-independent and associated labels using the text tool of the programming toolbar. To do so, click on the text tool.

 → **Text tool**

When this icon is active, open a text input box by clicking on a free area of the programming interface or on a block. After you have entered the label text, simply click anywhere outside the label window or press the [ESC] key. The window is closed and the label text is displayed in the diagram. That label can now be selected, moved or aligned.

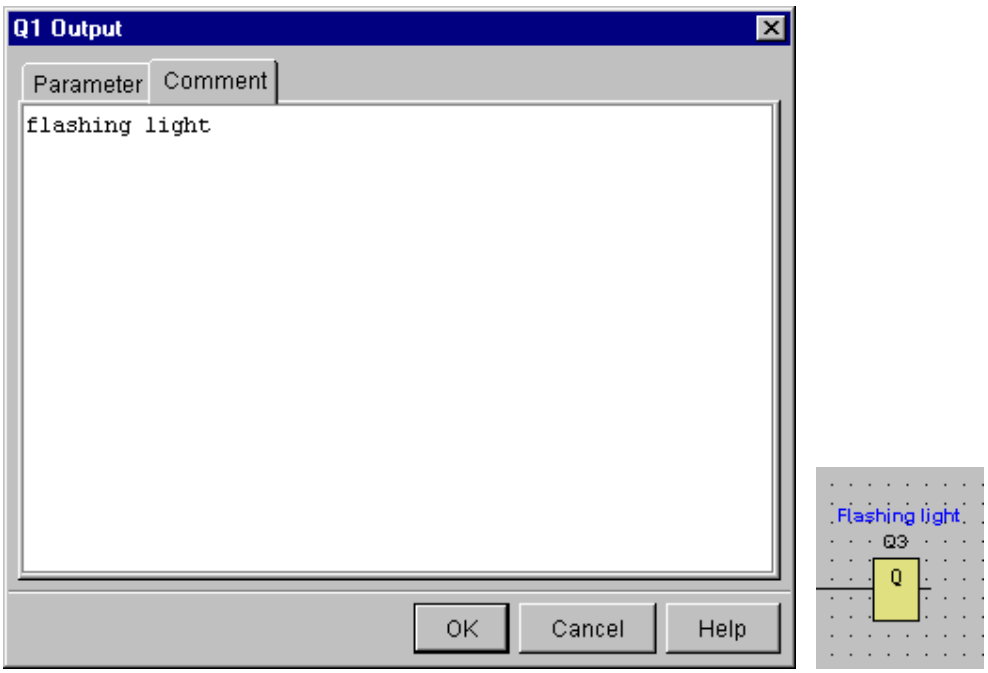
Example for the FBD Editor:



Block independent and associated text

Click on a free area of the programming interface to create a block independent label. A label can be edited by calling the text tool and then clicking on the relevant label.

By clicking on a block with the text tool, you create an associated label, namely the block comment. You can also input and edit this comment in the comments tab of the block properties dialog. The block comment can be used, for example, to assign a name to a block or to insert comments describing the task of the block within your circuit.



If you select a block with an associated label, the text is not marked. However, when you move the block, you also move the label. When you copy or cut the block, only the block itself is copied to the clipboard. A cut operation deletes the associated label. However, the associated label can be selected and moved, copied, cut or pasted individually. An associated label that is pasted from the clipboard is no longer associated with the block.

In Edit -> Input/Output Names you can assign block numbers and connector names to the I/Os.

3.2.4.2 Opening and saving the circuit program

Saving the circuit program

Click on the save icon in the standard toolbar to save the circuit program.

 → File Save

The circuit program is saved under the name it was opened with, while older versions are overwritten. When you initially save it, you are prompted to specify a program path and name.

Opening a circuit program

You can always clicking on the **File open** icon to open a circuit program for further editing. You open a list of recently opened programs by clicking on the arrow icon on the right side of the button.

 → File Open

3.3 Simulation of a circuit program

3.3.1 Starting the simulation

Use the Tools -> Simulation menu command or the simulation icon  in the programming toolbar to place your circuit program in simulation mode.

At the start of simulation, LOGO!Soft Comfort verifies the circuit program and shows any existing errors in the Info Window, which you can view by calling the View -> Info Window menu command or by pressing the function key [F4]. You can also use function key [F2] in the Info Window to display the LOGO! modules capable of running your program.

In simulation mode, you have the simulation toolbar and status window available for performing the simulation and for observing and controlling the behavior of your circuit program.

3.3.2 Layout of inputs

The inputs are displayed in the form of key or switch icons. The name of the input is displayed below the icon. An open input represents an inactive switch. When you click on the icon, it is indicated active and the switch is shown in closed state.

→ Icon for pushbutton I1, not actuated → open input

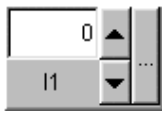
→ Icon for pushbutton I1, actuated → closed input

→ Icon for pushbutton I2, not actuated → open input

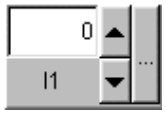
→ Icon for pushbutton I2, actuated → closed input

Layout of analog and frequency inputs

You can set the analog voltage or frequency values for analog and frequency inputs by means of a slide resistor. Click on the relevant block to pop up and operate this slide controller directly in the diagram. If you want to specify a more precise value, enter it directly or set it directly via the up/down keys at the side of the input window.



→ Display for input block I1 → Frequency input



→ Display for input block I1 → Frequency input

Input Functions

You set the response of your inputs for simulation purposes with the Tools -> Simulation Parameters menu command.

3.3.3 Layout of the outputs

In simulation mode, the outputs Q and the flag M. are displayed as outputs.

The status of an output or flag is indicated by a light or dark bulb icon. The name of the output in your circuit program is displayed below this icon.

 → Status display of output Q1 → Output switched off

 → Status display of output Q1 → Output switched on

The output status only indicates the status as such. Here, you cannot switch an output by clicking on an icon. When your circuit program switches an output, the indicator lamp is active; when the output is switched off, the indicator lamp is also switched off.

3.3.4 Set output

In simulation mode, you can select the command Set output by right clicking the mouse on the digital output of a block.

This command allows you to set an output, irrespective of the current status of a block. The output remains set until you enable it again or you end the simulation.

This way you can use a simulation to check how a circuit program will react to certain states.

3.3.5 Power failure

The user can simulate a power failure by clicking on the **Power** icon to interrupt the power supply to all inputs.

 → Power icon, not actuated

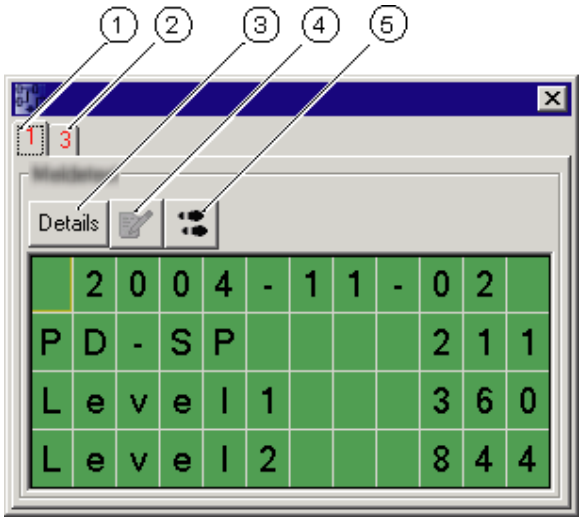
 → Power icon, actuated → Simulated power failure

This function can be used to test the reaction of the circuit to power failure and restart, as well as its retentivity. In contrast to the start of simulation, retentivity is relevant for the "Power failure" function. The start of a simulation is equivalent to the "Load Program" function in LOGO!. All values are reset, including the retentive values.

3.3.6 Layout of message texts

If you right-click on the entry in the message text, you can see from which block the entry in the message text originates. You can also select this block in the circuit program (**Go to Block**) and call up the properties of this block (**Block Properties**).

Standard View



① Tab of the displayed message text stating the priority.

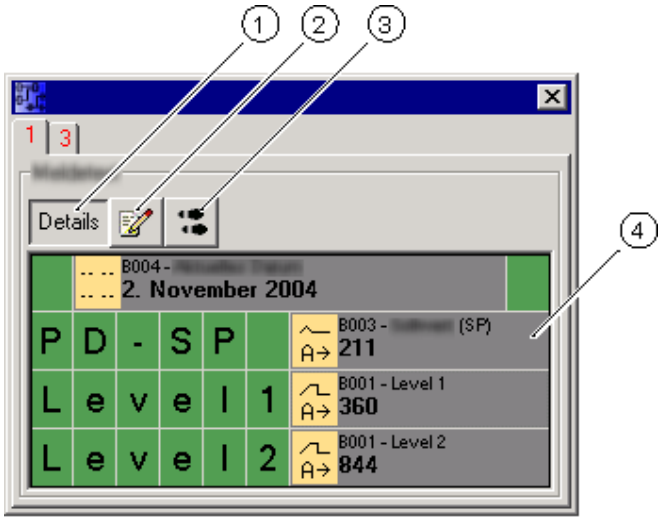
② Tab of another message text.
Here you can see that another message text of priority 3 exists.

③ **Details** button
If you click on this button, the view changes and you receive more detailed information in the Detail View (see below).

④ **Enter value manually** symbol
Before you can use this function you must first click on a changeable entry in the message text.
If you then click on this button, you can manually change the current value.
Alternatively you can also double-click on an entry in order to manually change it.

⑤ **Go to Block** symbol
If you click on this button, the special function belonging to the message text is selected in the circuit program.

Detail View



① **Details** button

If you click on this button, you return to the standard view (see above).

② **Enter value manually** symbol

Before you can use this function you must first click on a changeable entry in the message text.

If you then click on this button, you can manually change the current value.

Alternatively you can also double-click on an entry in order to manually change it.

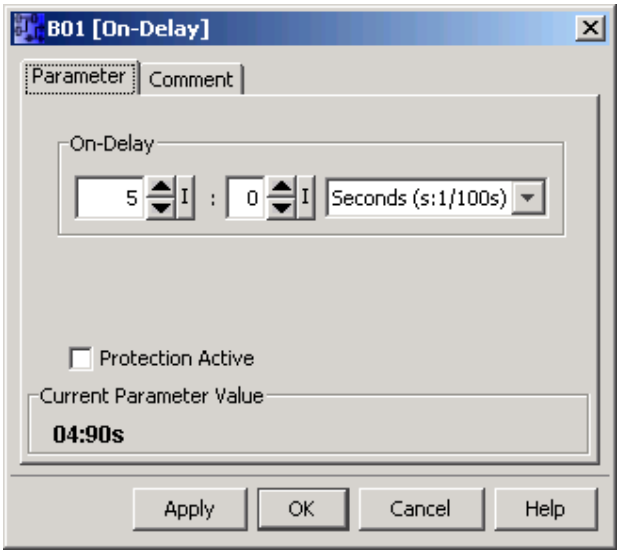
③ **Go to Block** symbol

If you click on this button, the special function belonging to the message text is selected in the circuit program.

④ Entry in message text with information regarding the block from which the entry originated.

3.3.7 Parameter assignment in simulation mode

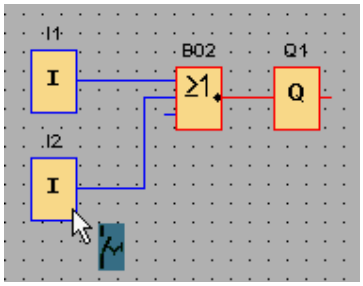
You can double-click on a block while a simulation is performed to open the block properties dialog. Here, as in programming mode, you can modify comments or parameters.



In simulation mode you are shown the actual parameter values. This analysis option allows you to test the reaction of your circuit program. You can open several parameter assignment windows concurrently in simulation mode.

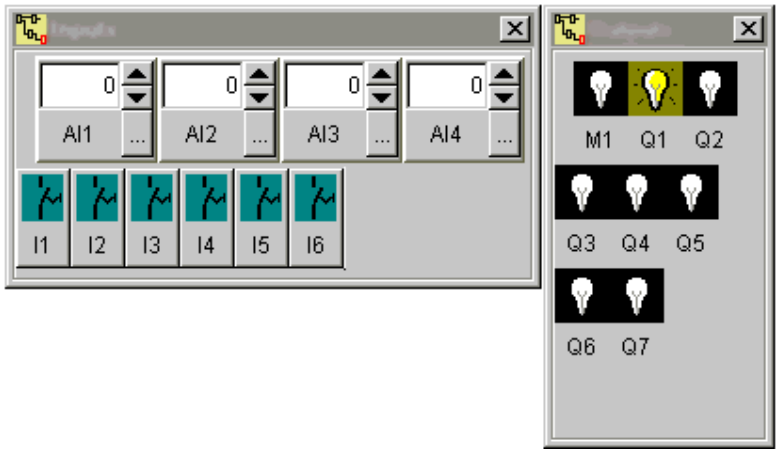
3.3.8 Alternative operation

You can click directly on the inputs to switch them on or off..



You can select the status window with the mouse, or drag and drop it out of LOGO!Soft Comfort to form a separate window. This is a particularly helpful feature for handling a large amount of I/Os in your circuit program, and for arranging the I/O layout to suit your requirements.

Example for the FBD Editor:



3.3.9 Controlling the simulation time

The simulation mode of LOGO!Soft Comfort allows you to test your circuit program on a timed basis, or over a specific number of cycles. You can even modify the time of day to test timer operations in your circuit program. See the time control section in the simulation toolbar topic for more information on the capabilities you have in simulation mode.

3.4 Practical example

3.4.1 Practical example: Introduction

This practical sample application for a service water pump offers newcomers a step-by-step introduction. In contrast to the previous tutorial, you learn here how to apply the functions that you have learned, based on a practical example.

Further samples of circuit programs are found in the sample applications section.

3.4.2 The task

Application

In addition to the drinking water supply, rainwater takes an increasing part in domestic water supply systems, thus saving money and helping to protect the environment. For example, rainwater can be used for the following needs:

- Washing clothes
- Watering the garden
- Watering indoor plants
- Washing the car
- Flushing the toilet

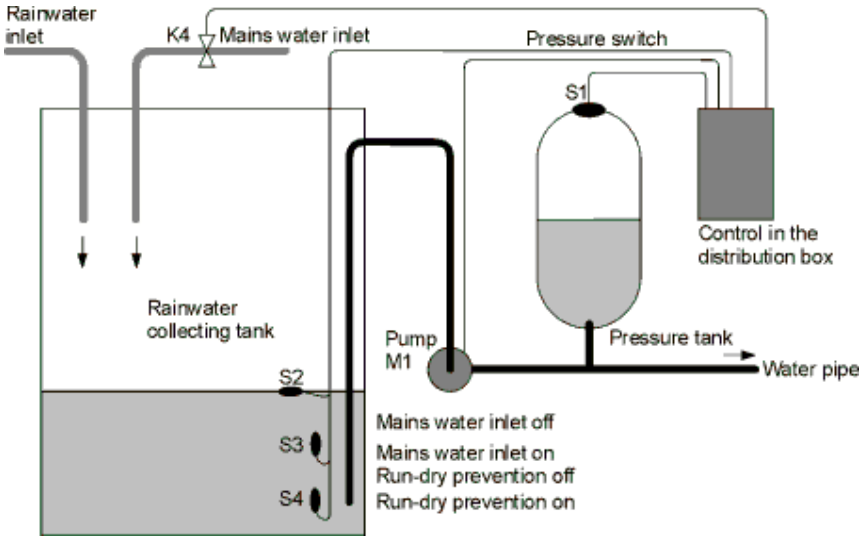
Rainwater can be collected in a suitable system to be used instead of drinking water.

Description of the system

The rainwater is collected in a reservoir. From the reservoir, it is pumped into a respective water supply system. From there it can be tapped in the same way as drinking water. The system functions can be upheld by supplying drinking water if the reservoir runs out of service water.

A control circuit is to be created that suits the requirements for such an installation.

The sketch below illustrates how such a service water system works:

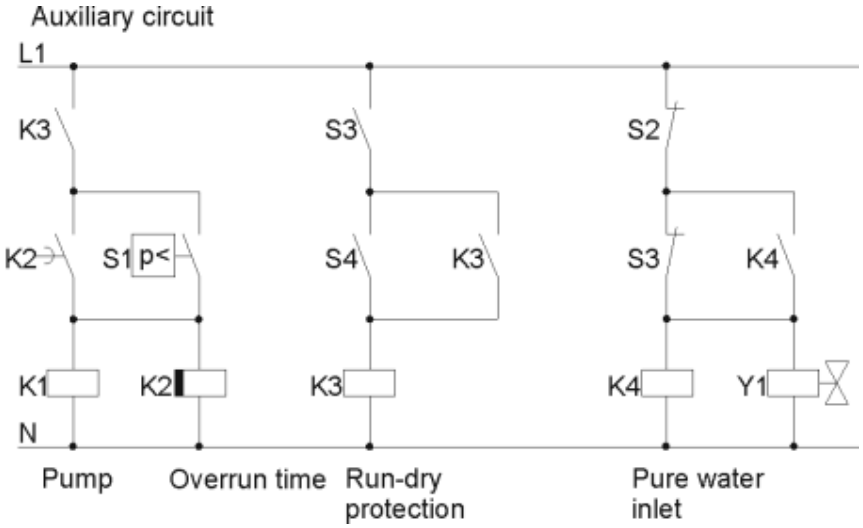


Requirements on the control system

- Service water must be available at all times. An emergency control system must change over to drinking water supply, for example, if service water runs low.
- The ingress of service water into the drinking water network must be prevented when switching over to drinking water supply.
- The pump must be disabled if the service water reservoir runs low of water (dry-run protection).

3.4.3 Layout of the solution

The pump and a solenoid valve are controlled by a pressure switch and three float switches, which are installed in the service water reservoir. The pump must be switched on if the pressure in the reservoir drops below minimum. When the operating pressure is reached, the pump is switched off again after a tracking time of a few seconds. The tracking time prevents pump oscillation if water is tapped over an extended period.

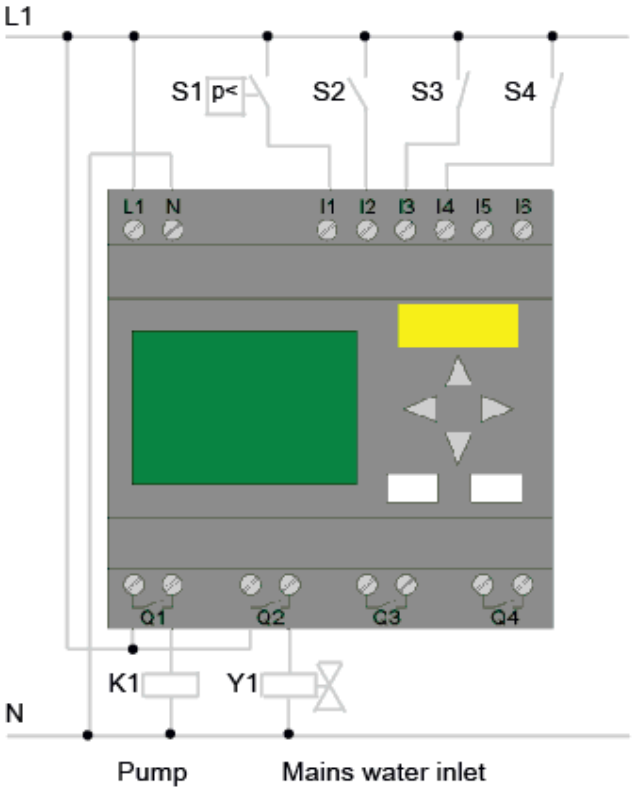


3.4.4 Solution with LOGO!

Connecting field devices

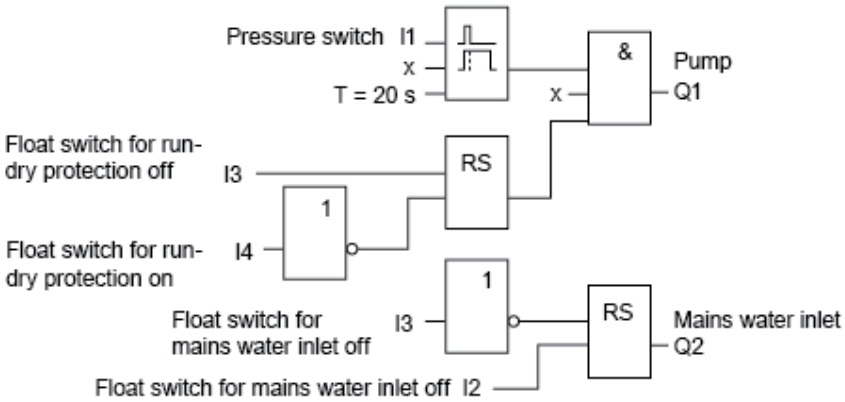
In addition to the LOGO! device, all you need is the pressure switch and the float switches to control the pump. If you are using a 3-phase AC motor, a contactor relay is required for switching the pump. In systems with single-phase AC pumps, you must install a contactor relay if the current of the AC pump exceeds the switching capacity of the output relay Q1. A solenoid valve can usually be controlled directly, due to its low power consumption.

Connection diagram



Block diagram

The block diagram shows you how to interconnect the pump controls and the solenoid valve. Its layout corresponds with the structure of the circuit diagram.



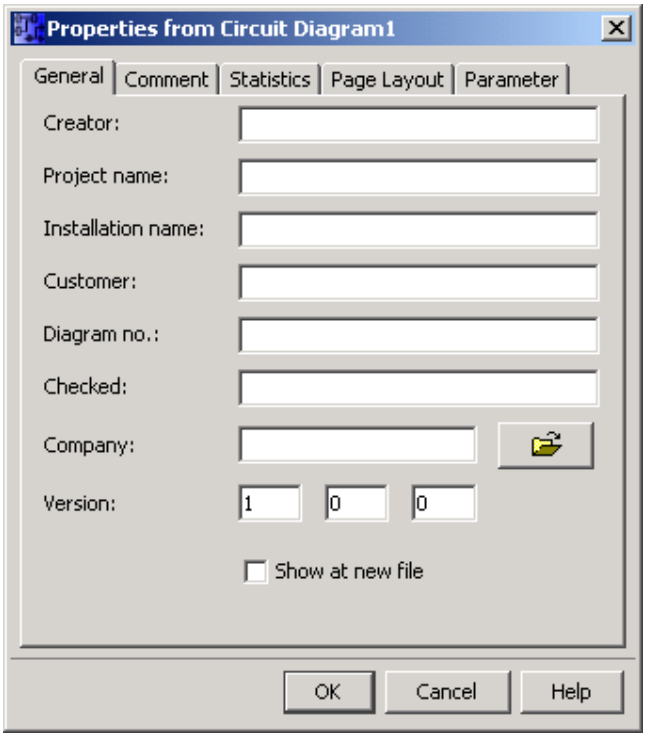
Options

For specific applications, you also have the choice of integrating additional functions which could otherwise only be realized with additional switchgear:

- Enabling the pump at specific times, for example, only during the summer months or at a specific time-of-day
- Indication of imminent or existing shortage of water
- System error messages

3.4.5 Input of project data

After you have planned your project, you can start to create it in LOGO!Soft Comfort. If you do not want to start programming right away, you can first input your project data in the properties menu dialog.

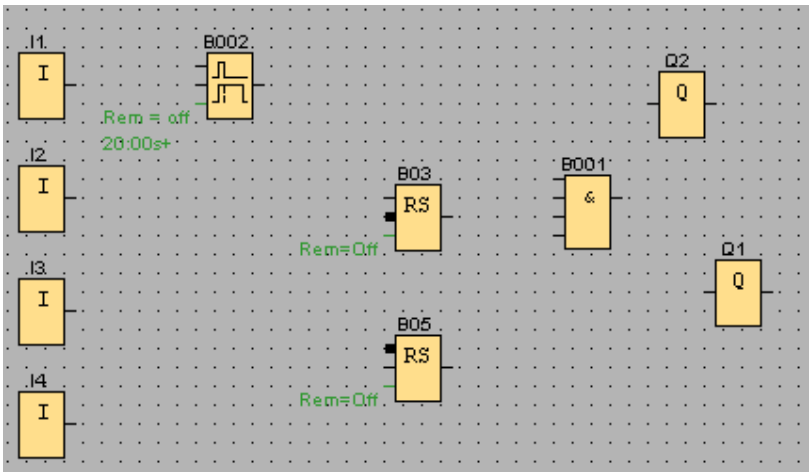


When working through the following steps, you should remember to save your circuit program at regular intervals. You may want to open a revised version at a later time to test out other options.

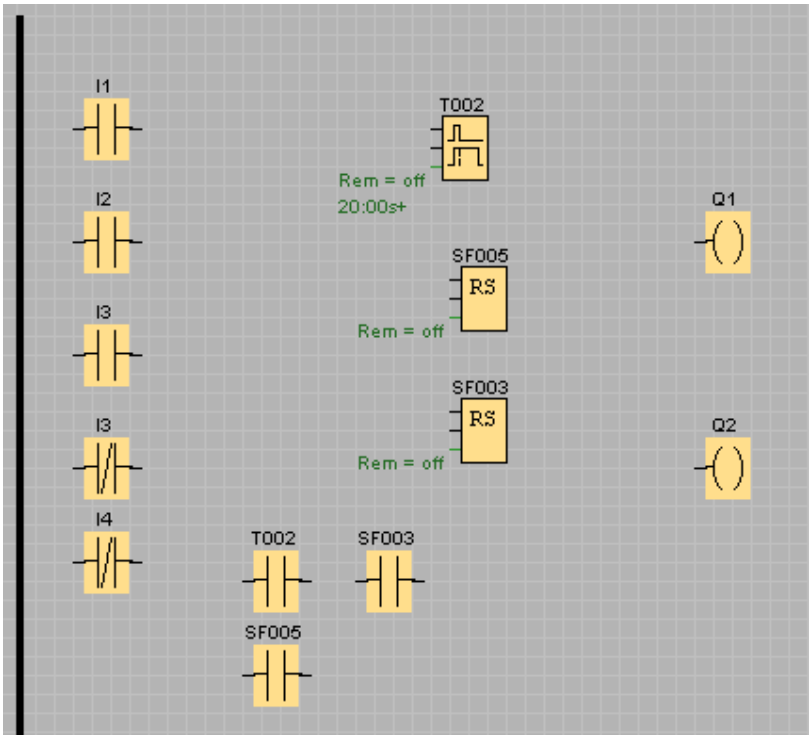
3.4.6 Placing blocks

In the next step, place the required blocks into the programming interface. Note that in addition to the standard and special functions, you also require I/O blocks. At this stage of progress it is sufficient for you to place the blocks roughly into position that seems appropriate to you for connecting them later. Fine positioning is carried out after all connections are made.

FBD Editor:



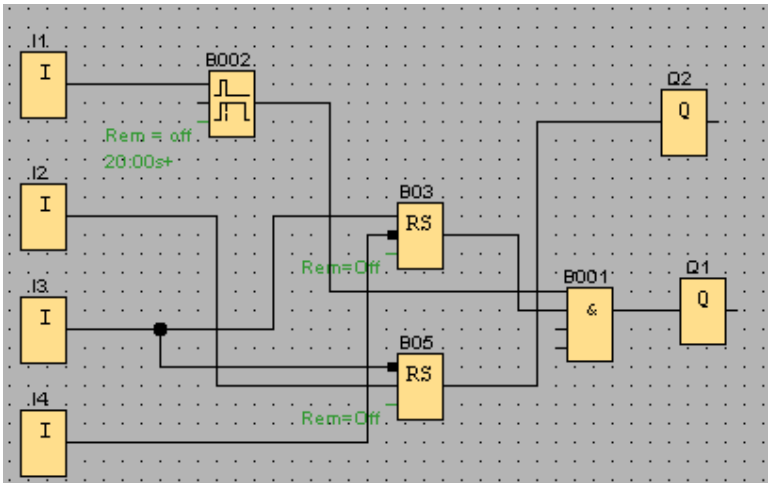
LAD Editor



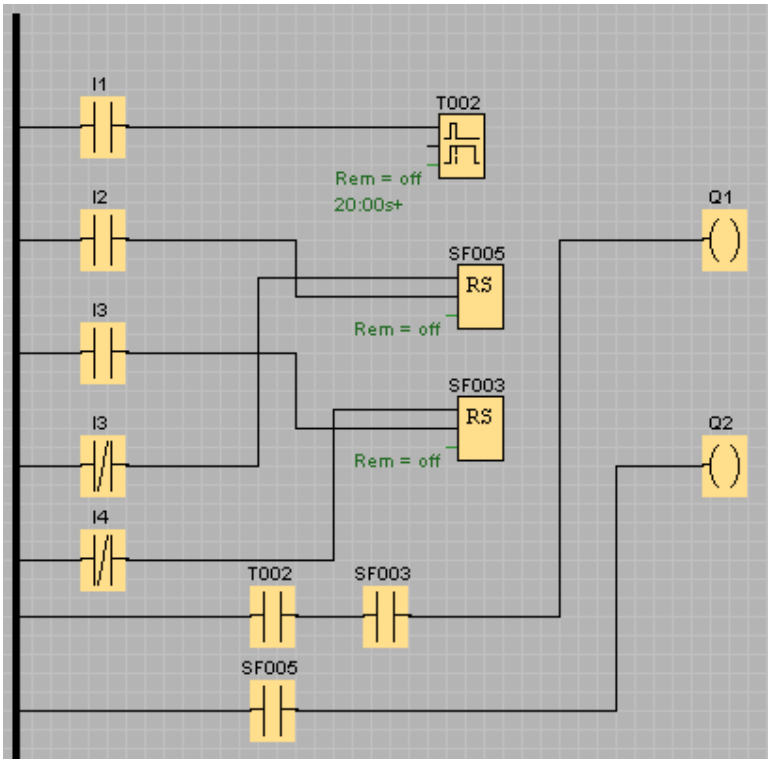
3.4.7 Connecting blocks

Connect the blocks as you have planned previously. Connect them by drawing the connecting line, starting at the output of a source block and ending at the input of the target block. This has the advantage that you are shown the name of the connector when you interconnect the input, which is particularly advantageous for the various connectors of SFBs.

FBD Editor:



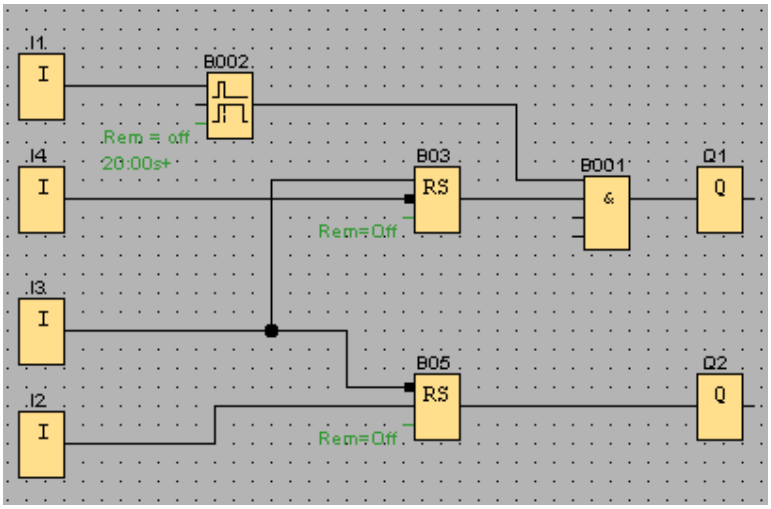
LAD Editor:



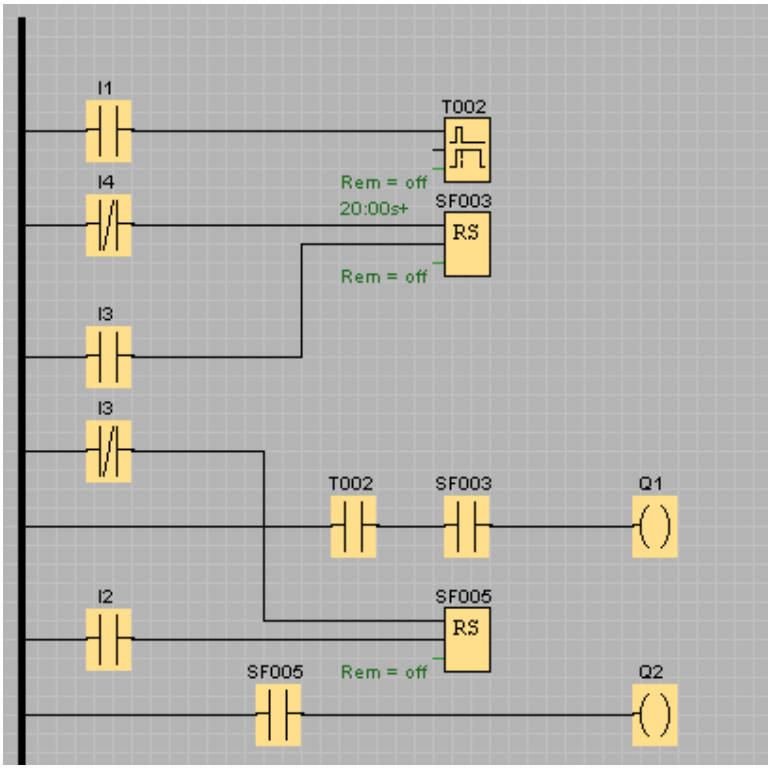
3.4.8 Cleaning up the programming interface

Some of the connecting lines may be routed across blocks. The layout of the circuit program is not particularly clear at the present. In order to "tidy up" the programming interface, select the connecting lines and blocks where required and move or align them until you have optimized the circuit program layout as far as possible.

FBD Editor:



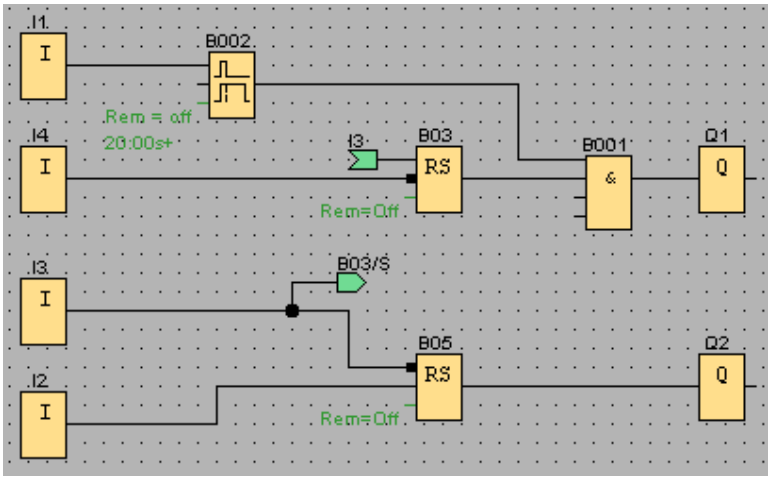
LAD Editor:



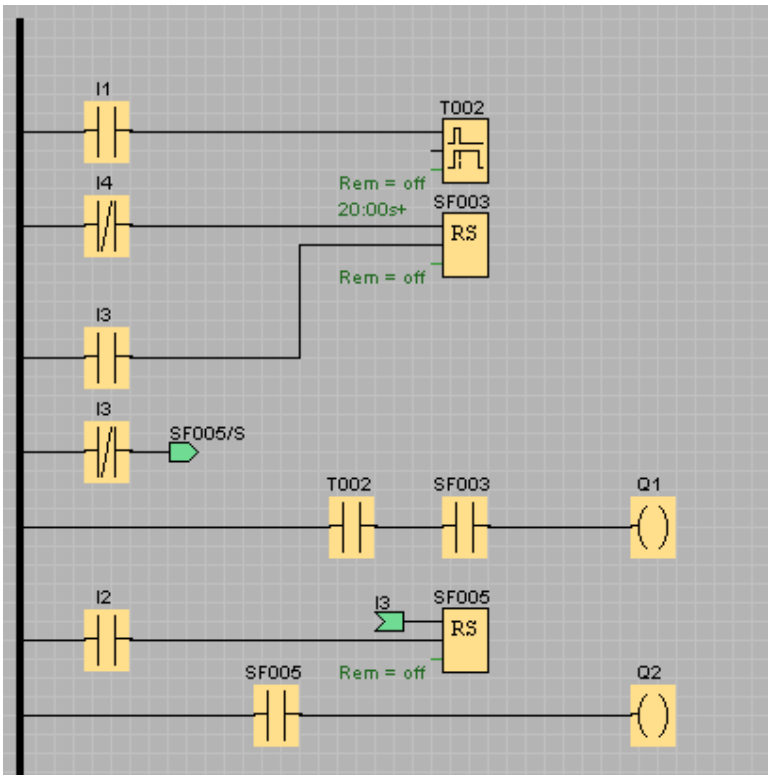
3.4.9 Optimizing the view

Unwanted though unavoidable line crossings can be cut using the Cut/Join tool. This improves the overview.

FBD Editor:



LAD Editor:



You have now completed your circuit program. Verify all connections once again and configure the block parameters to suit your requirements.

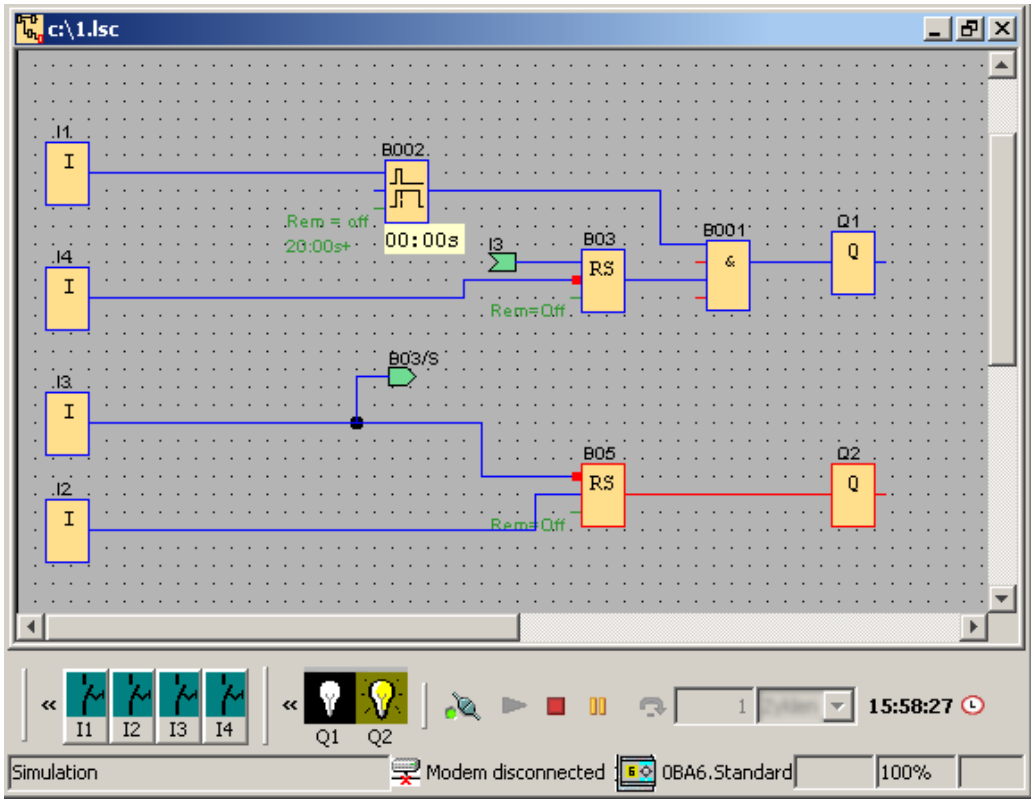
3.4.10 Testing the circuit program

Place your circuit program in simulation mode and verify that your program is error-free.

Although you now know that you can run your circuit program in LOGO!, you still need to make sure your circuit program operates as planned. You may also want to modify certain parameters. You can try out different input values, test the reaction of the system to power failure and compare your calculations or expectations with the simulated reaction of the outputs. The simulation toolbar and status window provides the tools you need for these tasks.

The float and pressure switches have a momentary action. If, however, you wish to simulate your circuit for testing purposes, simply change the input function from momentary action to switching action.

Example for the FBD Editor:



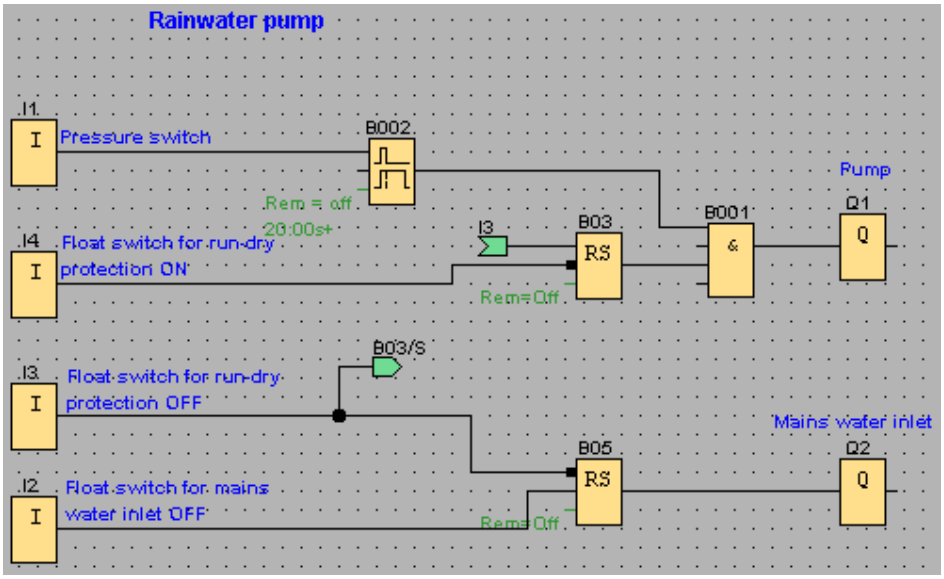
When you have made all corrections and everything runs as expected, you can start to document your circuit program.

3.4.11 Circuit program documentation

Program comments

You can now start to add comments to your program using the Text tool. Describe the I/Os to make your circuit program easy to understand. You do not have to display the connector names on-screen. You should nevertheless assign names to the connectors, because you may want to print out a connection list at a later time. From the comment tab of the File -> Properties dialog, you can add a comment to your circuit program, which you can include when you print the circuit program.

Example for the FBD Editor:



Saving the file to a storage medium

Before you transfer your circuit program, you should save it once again. Choose the relevant command from the menu and enter a program name and path.

Printing the circuit program

Use the FIle->Print command to print a copy of your circuit program. From this dialog you specify the print format and the details to be included or omitted.

3.4.12 Transferring the circuit program

Password protection

To protect your process solution and prevent unauthorized access to your circuit program, you can assign a password before you transfer it to LOGO!.

To assign this password, use the Parameter tab of the File->Properties menu command dialog. Enter your password and confirm it with OK.

Password protection is used when you transfer the circuit program to LOGO! and is activated when LOGO! exits transfer mode.

The password protects your circuit program in LOGO!. Editing values and parameters, or viewing the circuit program in LOGO!, or uploading the circuit program from LOGO! to the PC is now only possible after you have entered the password.

Transferring the circuit program

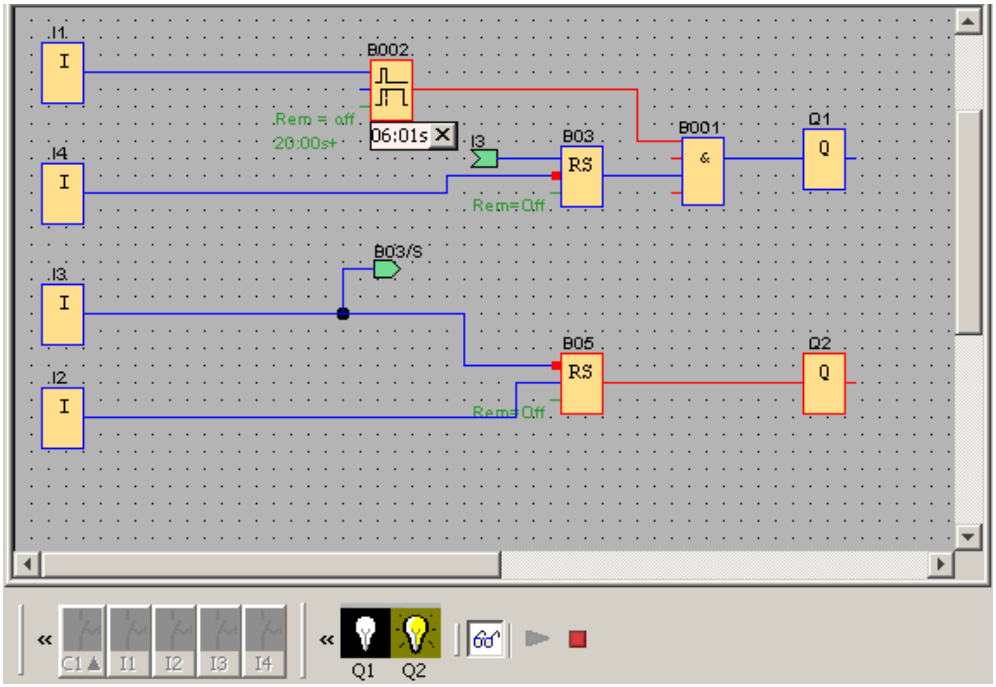
Finally, transfer your circuit program to a suitable LOGO! version and then connect the module. Connect LOGO! with the consumer devices in your project.

You have created the circuit program under LOGO!Soft Comfort within a very short time and, compared with conventional methods, you have saved yourself a considerable amount of time and effort.

3.4.13 Performing an online test of the circuit program

After you are satisfied with your program simulation and have downloaded it to LOGO!, you can also perform an online test of the circuit program. An online test is similar to simulation in that you can view inputs and outputs and block parameters. It differs, however, in that you are testing the program running in the LOGO! device with "live" inputs rather than testing the program on the PC with simulated inputs.

The following example shows an online test of the practical example circuit program. In this example, input I1 was turned on and then turned off. The off-delay timer began counting when I1 was turned off:



See the topic Tools->Online Test for prequisite conditions for an online test.

4 Sample applications

4.1 Sample applications - Overview

Introduction

To give you an impression of the versatility of LOGO!, LOGO!Soft Comfort includes a small collection of applications, in addition to the service water pump application shown in the tutorial.

This online help briefly describes the tasks and presents the relevant solution with LOGO!Soft Comfort. These circuit programs, as well as many others, are found on your LOGO!Soft Comfort CD-ROM in the ..\Samples folder. You will also find there documentation for the various samples.

Additional sample programs are also available on the Internet at http://www.siemens.com/logo. Select "Products and Solutions", then "Applications".

Note

LOGO! sample applications are available free of charge to our customers. These are provided without guarantee, and are intended for general information about the possible fields of application for LOGO! modules and LOGO!Soft Comfort software. Custom solutions may be different.

The user operates the system at his own responsibility. We also refer to local standards and system-related installation regulations.

This section presents the following sample applications:

- Air-conditioning system
- Factory door
- Heating control
- Fill station

Please also note the service water pump example.

4.2 Air-conditioning system

Requirements for an air-conditioning system

An air-conditioning system supplies fresh air into a room and exhausts the contaminated air. Let us look at the following sample system:

- A room contains an extractor fan and a fresh-air fan.
- Each fan is monitored by means of a flow sensor.
- The pressure in the room may rise above the atmospheric pressure.
- The fresh-air fan may only be switched on if the flow sensor signals the safe operational state of the extractor fan.
- A warning lamp indicates failure of one of the fans.

Standard solution

The fans are monitored by means of flow sensors. If no air flow is registered after a short delay time has expired, the system is switched off and an error message is generated, which can be acknowledged by pressing the off button.

Fan monitoring requires an analyzer circuit with several switching devices, in addition to the flow sensors. A single LOGO! device can replace this analyzer circuit.

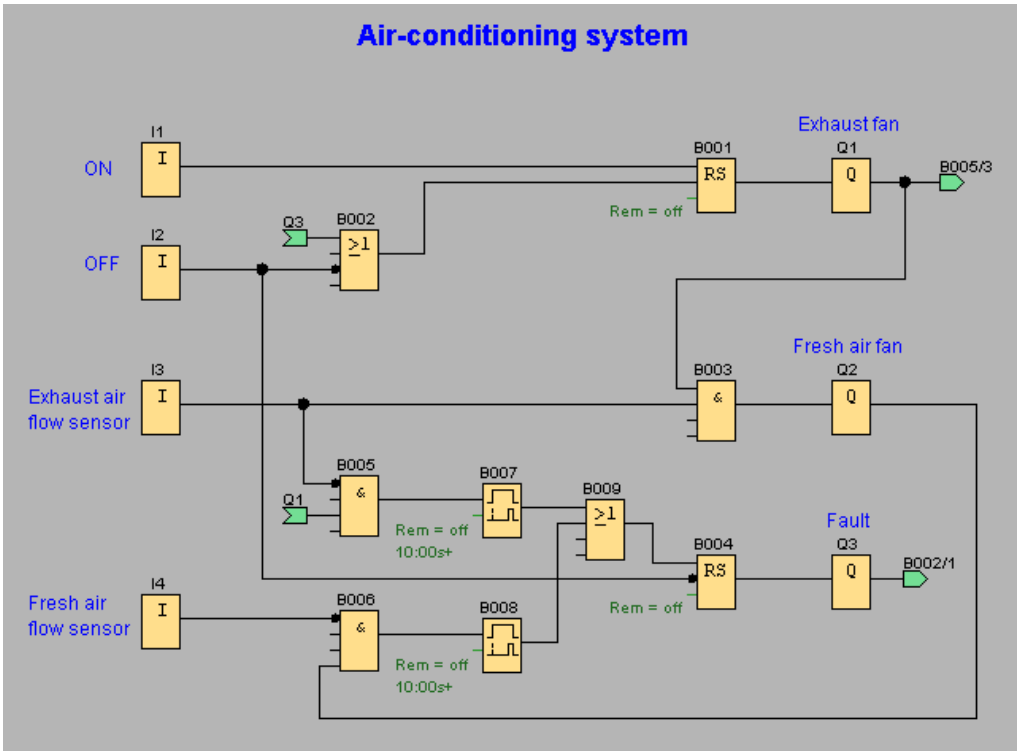
LOGO!Soft Comfort solution

The use of LOGO! reduces the amount of switchgear. Thus, you save installation time and space in the control cabinet. You may even be able to use as a smaller control cabinet.

With LOGO! you can also switch off the fans sequentially after the system is switched off.

The circuit in LOGO!Soft Comfort

The system is switched on and off at the inputs I1 and I2. The fans are connected to outputs Q1 and Q2, the flow sensors are connected to the inputs I3 and I4. Blocks B007 and B008 are used to set the watchdog times after which the flow sensors should send a signal to the fault output Q3.



You can invert output Q3 to use output messages at Q4. Relay Q4 only drops out if main power is lost or if there is a fault in the system. The output can then be used for a remote message.

4.3 Factory door

Requirements for a gate control system

In many cases a factory entrance is closed with roll gates. Those gates are only opened when vehicles need to enter or leave the factory grounds. The gate is controlled by a gatekeeper.

- The sliding gate is opened and closed by means of a pushbutton control in the gatehouse. The gatekeeper can monitor the gate operation.
- The roll gate is normally fully opened or it is closed. However, gate movements can always be interrupted.
- A flashing light is activated five seconds before the gate moves and while the gate is in motion.
- A safety pressure strip ensures that people are not injured and that no objects are trapped and damaged when the gate is closing.

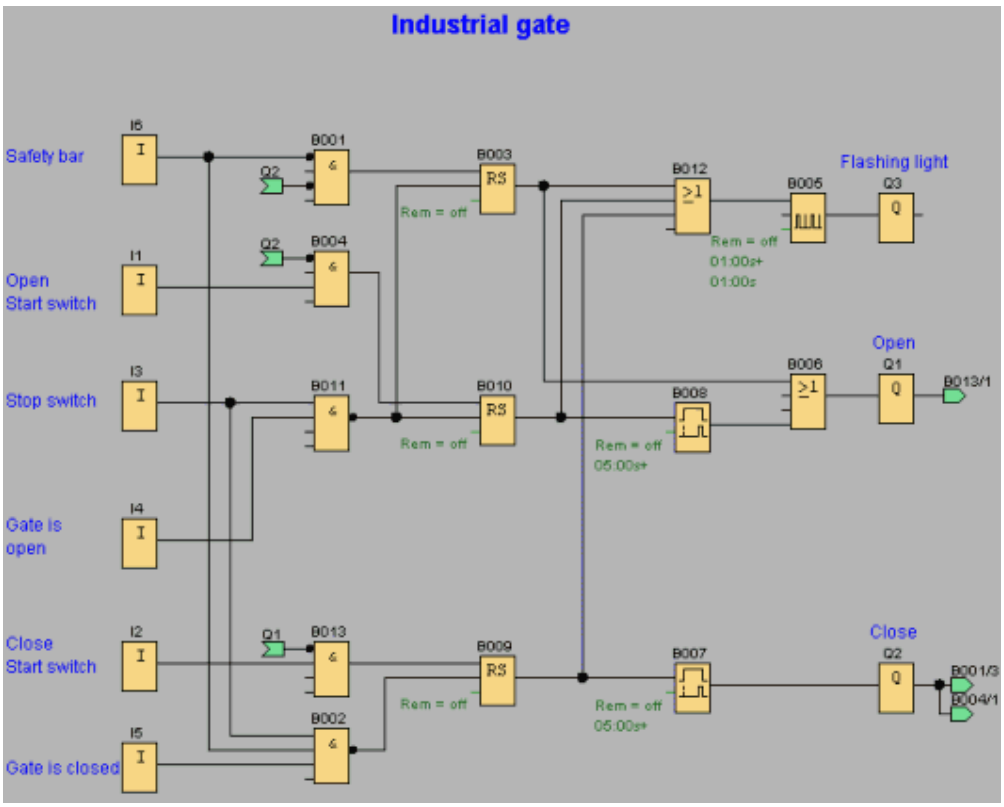
Standard solution

There are many different control systems for operating automatic gates. The OPEN and CLOSE buttons initiate gate movements in the relevant direction, provided it is not already moving in the opposite direction. Movement of the gate is terminated either by means of the STOP button or the relevant limit switch.

LOGO!Soft Comfort solution

A LOGO! circuit provides a further feature compared to standard controls: The actuation of a safety bar interrupts the closing motion of the gate. Five seconds before the gate opens or closes, a flashing light is activated and signals the start of the movement. It continues flashing until the gate has stopped.

In contrast to standard solutions, LOGO! offers an easy and economic means of modifying the control system.



4.4 Heating control

Demands on the heating control

The example illustrates the counter rotational nature of lead temperature and outdoor temperature with a heating control.

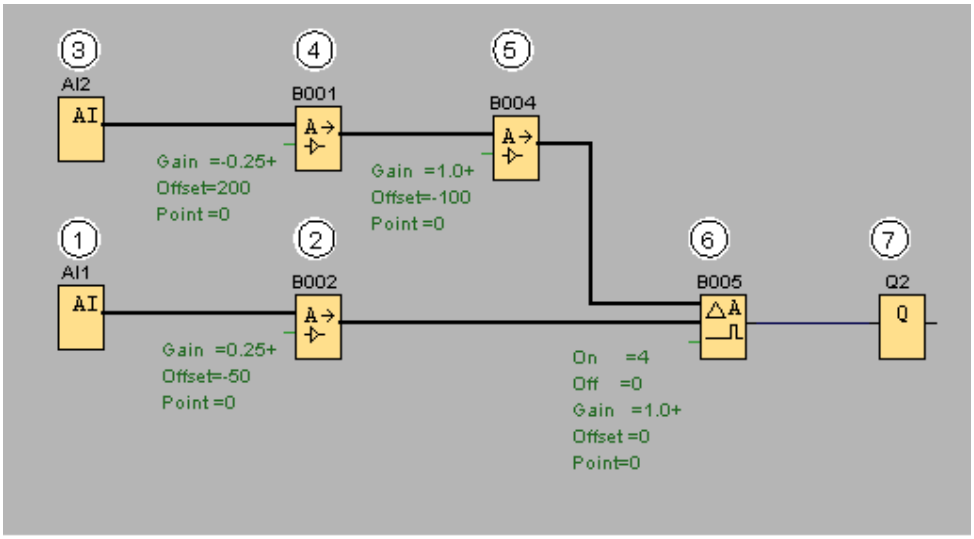
The lead temperature of the heating should be controlled inversely proportional to the outdoor temperature. This means: The lower the outdoor temperature, the greater the lead temperature.

Outdoor and lead temperatures are measured using PT100 sensors.

With an outdoor temperature of 0 °C, the lead temperature (**x**) should be 50 °C.

If the outdoor temperature drops by more than 4 °C, the heating should switch on.

LOGO!Soft Comfort solution



(1) A PT100 sensor is connected to an AI1, and this measures the lead temperature.

(2) The analog amplifier is parameterized as follows:

- Sensor: PT100 (proportional)
- Measuring range and parameters are stipulated by the PT100 sensors.
- Unit: Celsius
- Resolution: x 1

The amplifier causes the actual temperature that has been measured by the sensor to be issued on its output.

(3) A PT100 sensor is connected to an AI2, and this measures the outside temperature.

④ The analog amplifier is parameterized as follows:

- Sensor: PT100 (**inversely** proportional)
- Measuring range and parameters are stipulated by the PT100 sensors.
- Unit: Celsius
- Resolution: x 1

The amplifier causes a value that is inversely proportional to the temperature measured by the sensor to be issued on its output.

Hence: The greater the outside temperature, the lower the issued value.

⑤ The analog amplifier is parameterized as follows:

- Sensor: No sensor
- Gain: 1
- Offset: -100 (**y**)

The value issued by ④ is edited (standardized) by this analog amplifier in such a way that it can be compared with the lead temperature.

⑥ The analog comparator is parameterized as follows:

- Sensor: No sensor
- Gain: 1
- Offset: 0
- Threshold value in: 4
- Threshold value out: 0

The analog comparator switches on the output Q2 ⑦ if the difference between the lead temperature and the standardized outdoor temperature exceeds 4 °C.

If the difference falls short of 0 °C, the analog comparator switches the output Q2 back off again.

⑦ Output Q2 switches the heating on and off.

Mode of operation

The outside temperature drops; this causes the value issued on the analog amplifier ⑥ to increase to the same extent. The difference on the analog comparator between the lead and the outside temperature increases.

If the difference exceeds 4 °C the heating is switched on.

By switching on the heating the lead temperature increases. Because of this, the difference on the analog comparator between the lead and the outside temperature lowers (provided the outside temperature drops more slowly than the lead temperature increases).

If the difference falls short of 0 °C, the heating is switched off.

Changing parameters

The Offset (y) parameter with the analog amplifier ⑤ depends on your desired lead temperature (x) with 0 °C outside temperature. The parameter is calculated as follows:

y = x -150

Furthermore, the switch-on threshold and the switch-off threshold of the heating can change by means of the threshold value of the analog comparator ⑥.

Note

You can save block ⑤ if you adapt the threshold value in the analog comparator accordingly.

Try it out!

The example can be found as a circuit program on the LOGO!Soft Comfort CD-ROM. Load the circuit program in LOGO!Soft Comfort and try out the instructions above in simulation mode.

4.5 Fill station

Requirements for a fill station

A box shall be filled with two different items, up to a specified total for each item. When all items are in the box, it will be transported to the packaging station. A conveyor belt transports the items of both types to the box. (This example does not show the filling conveyor belt.) This circuit program for this example uses two up/down counters to count the items of each type, an analog math instruction to sum the total number of items, and message texts to be displayed on the LOGO! Display and LOGO! TD (Text Display) that show the number of items of each type and the total number counted so far.

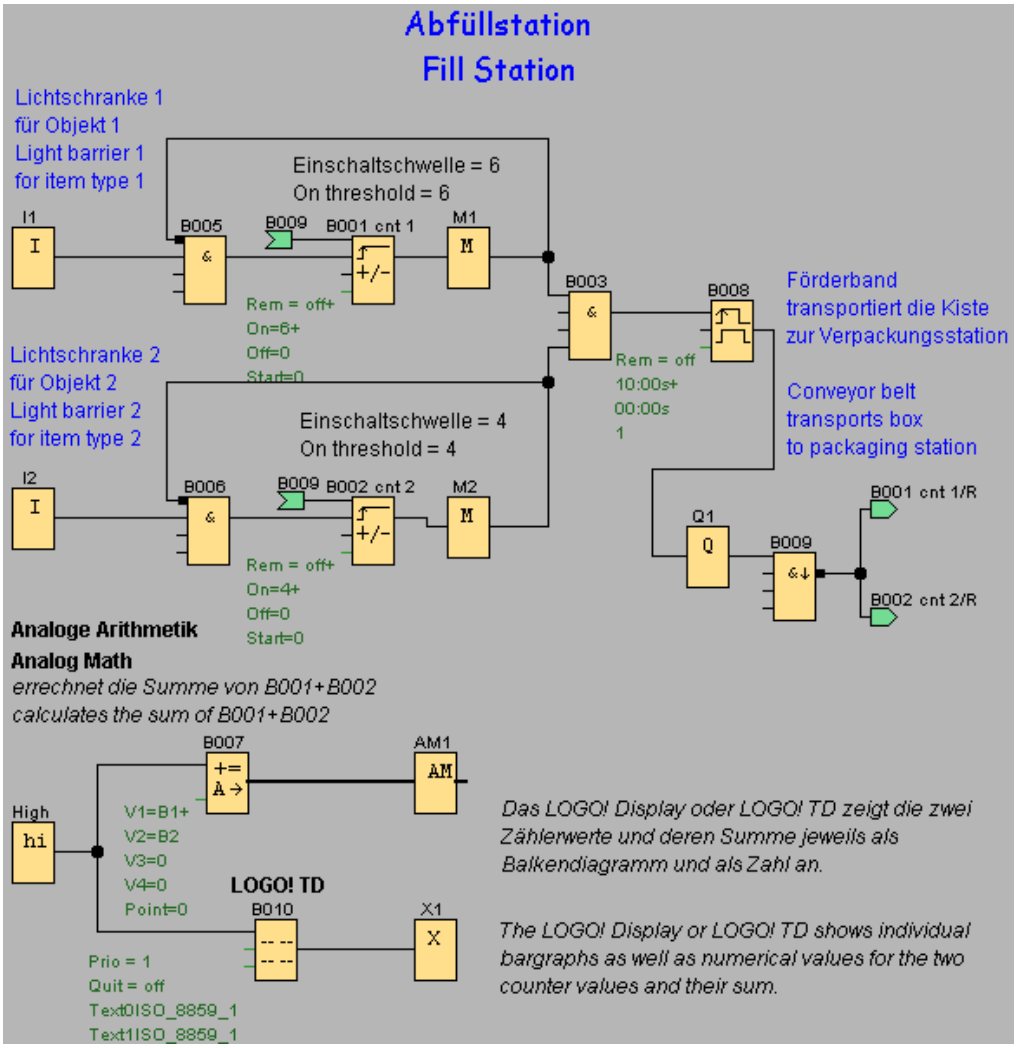
This process is described below:

- To fill the box, each item is transported by a conveyor belt to the box (not part of this example).
- The sequence in which the items fall into the box is random.
- Each item that falls into the box is counted by a sensor.
- The connected LOGO! TD as well as the LOGO! Display must display how many items of each kind have been counted, how many total items are in the box.

The circuit program in LOGO!Soft Comfort

Light barriers are connected to the two digital inputs I1 and I2 sensors that detect when an item falls into the box. The two counters (B001 and B002) count each item for the two specific types of items as they fall into the box. The on-threshold of each counter specifies the maximum number possible for each item type. When the box is full, a conveyor belt activates for ten seconds to transport the filled box to the packaging station and to transport an empty box to the fill station.

The circuit program uses a message text function block to display on the LOGO! TD and LOGO! Display how many total items and how many of each type have been counted. The message text uses the “ticking” feature to alternate between displaying bar graphs of the counted items and displaying a text summary of the counts. In addition, the text part of the message text will be shown in either English or German, depending on the current character set configuration for message texts.



Parameters of the functions blocks

Note the following parameter usage:

- The on-thresholds of counters B001 and B002 define the maximum number of items of each item type to be put in the box.
- The pulse width (TH) of the edge-triggered wiping relay defines the duration of the movement of the conveyor belt.
- The Analog Math function block B007 that calculates the total is programmed as follows:
- Value 1 is a reference parameter to the current value of counter B001.
- Value 2 is a reference parameter to the current value of counter B002.
- Operator 1 is "+" to sum the two counters.
- Priority 1 is high. (This is the only operation in the function block, so the priority in this case is irrelevant).
- The remaining operands and operators are set to "+ 0" and do not affect the output of the equation.

Message texts

Message text block B010 displays a message text whose four lines all tick “line by line”, such that two screen forms display in alternation. The first shows bar graph representations of the counted items (B001 and B002 current values) and a bar graph for the summed total items as calculated by B007. The second screen form shows the same values numerically and with text descriptions.

The message text function block is configured with two character sets enabled. The message text for character set 1 is configured with English text and the message text for character set 2 is configured with German text. You use the File -> Message Text Settings to select the two character sets. In this case you would select the same character set for both English and German characters. When LOGO! is in run mode, the current character set selection for message texts determines which message will display.

5 Reference material

5.1 Constants and connectors

5.1.1 Constants and connectors - Overview

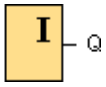
This tool must be selected if you want to place input blocks, output blocks, flags or constants (high, low) on the programming interface. The specific type of block to be inserted is selected from an additional toolbar that pops up when you select the **Constants and Connectors** tool.

Display in the FBD Editor	Display in the LAD Editor
I Inputs	Make contact
C Cursor keys	Analog contact
F LOGO! TD function keys	Break contact
S Shift register bits	Relay coil
lo hi Permanent logical levels	Inverted output
Q Outputs	
X Open connectors	
M Flags	
AI Analog inputs	
AQ Analog outputs	
AM Analog flags	

The number of available icons depends on the LOGO! version you have selected.

5.1.2 FBD

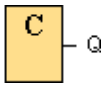
5.1.2.1 Inputs



Input blocks represent the input terminals of LOGO!. Up to 24 digital inputs are available to you.

In your block configuration, you can assign an input block a new input terminal, if this terminal is not already used in the circuit program.

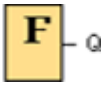
5.1.2.2 Cursor keys



Up to four cursor keys are available to you. Cursor keys are programmed for the circuit program in the same ways as other inputs. Cursor keys can save switches and inputs, and allow operator control of the circuit program.

The four cursor keys on the LOGO! TD are the same cursor key inputs on the LOGO! Basic module. Pressing ESC + C4B, for example, on either the LOGO! TD or LOGO! activates a single input for C4B.

5.1.2.3 LOGO! TD function keys



The LOGO! TD module has four function keys that you can use as digital inputs in your circuit program. You program the function keys in the same way as other inputs in your circuit program. Function keys can save switches and inputs, and allow operator control of the circuit program.

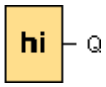
5.1.2.4 Outputs

Output blocks represent the output terminals of LOGO!. You can use up to 16 outputs. In your block configuration, you can assign an output block a new terminal, provided this terminal is not already used in your circuit program.

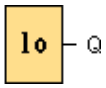
The output always carries the signal of the previous program cycle. This value does not change within the current program cycle.

Overview

5.1.2.5 Permanent logical levels

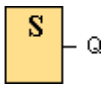


Set the block input to logical **hi** (hi = high) to set it permanently to logical ‘’or 'H' state.



Set the block input to logical **lo** (lo = low) to set it permanently to logical ‘’or 'L' state.

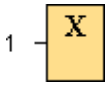
5.1.2.6 Shift register bits



LOGO! provides the shift register bits S1 to S8, which are assigned the read-only attribute in the circuit program. The content of shift register bits can only be modified by means of the Shift register special function.

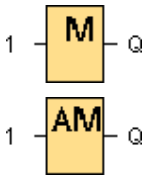
Overview

5.1.2.7 Open connectors



Interconnect the output of an unused block (for example, message texts) with the "open connector" block.

5.1.2.8 Flags



Flag blocks output their input signal. LOGO! provides 27 digital flags M1 ... M27 and 6 analog flags AM1 ... AM6.

0BA4, 0BA5: 24 digital flags M1 ... M24; 6 analog flags AM1..AM6
0BA3, 0BA2: 8 digital flags M1 ... M8
0BA1: 4 digital flags M1 ... M4
0BA0: 0 flags

In your block configuration, you can assign a new number to the flag, provided this flag number does not already exist in your circuit program.

The output always carries the signal of the previous program cycle. This value does not change within the current program cycle.

Startup flag: M8

The M8 flag is set in the first cycle of the user program and can thus be used in your circuit program as a **startup flag**. It is reset after the first program execution cycle.

In the subsequent cycles, the M8 flag reacts in the same way as the M1 to M7 flags.

Backlight flags: M25 and M26

The M25 flag controls the backlight of the LOGO! Display. The M26 flag controls the backlight of the LOGO! TD (Text Display).

You can use the outputs of timers, message texts, or other function blocks to activate the backlight flags. To enable multiple conditions to control the backlight of the devices, you can use multiple function blocks in parallel or in sequence.

Message text character set flag: M27

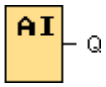
The M27 flag, if used, determines whether the message texts of the primary character set or the secondary character set will display. You select the two character sets from either the Msg Config menu of LOGO! or the File -> Message Text Settings menu command of LOGO!Soft Comfort. Then when you configure message texts, you select whether a particular message text consists of characters from the primary character set (Character Set 1) or the secondary character set (Character Set 2).

In the circuit program, M27 can be used to enable the message texts of either the primary or secondary character set and to disable the message texts of the other. When M27=0 (low), then LOGO! only displays the message texts from the primary character set. When M27=1 (high), then LOGO! only displays the message texts from the secondary character set.

Analog Flags: AM1 to AM6

You can use the analog flags serve as a markers for analog inputs or analog instruction blocks. The analog flag merely accepts an analog value as input and outputs that value.

5.1.2.9 Analog inputs



The LOGO! versions 12/24 RC, 12/24 RCo, 24 and 24o, as well as the expansion modules AM2 12/24 process analog signals. You can use up to eight analog inputs. In your block configuration, you can assign a new input terminal to an input block, provided this terminal is not already used in the circuit program.

Some of the input terminals of the LOGO! 0BA6 versions 12/24 RC, 12/24 RCo, 24 and 24o have a dual definition: they can be used as either digital inputs or analog inputs. See the LOGO! manual or product information for specific information about specific modules. Circuit programs written for these modules for the 0BA5 release and earlier can run on 0BA6 modules without modification. New circuit programs can make use of the new input features, which provide additional fast-speed counters and analog capability.

For help on analog block parameters, refer to Information on analog value processing.

The block input number is not determined by the hardware structure in systems operating with devices of the 0BA0 to 0BA2 series.

Overview

5.1.2.10 Analog outputs

Two analog outputs are available, namely AQ1 and AQ2. You can only set an analog value at the analog output, that is, a function with an analog output or analog flag AM.

If you connect a special function (that has an analog output) to a **real** analog output, then note that the analog output can only process values from 0 to 1000.

As of the 0BA5 device series, you can configure the behavior of analog outputs in STOPmode. Analog outputs can retain their last values when LOGO! goes to STOP mode. Alternatively, you can configure specific values to be set for AQ1 and AQ2 when LOGO! Goes to STOP mode.

You can also set the analog output value range. You have two choices:

- Normal (0 to 10 V or 0-20 mA)
- 4-20 mA

0BA0 to 0BA5: You cannot configure the behavior of AQ1 and AQ2 on transition to STOP mode for LOGO! modules prior to the 0BA5 device series.

5.1.3 LAD

5.1.3.1 Break contact

Break contacts, as well as make contacts and analog contacts represent the input terminals of a LOGO! module.

A pop-up window opens when you insert the contact in your circuit diagram. In this dialog you can specify the type of input according to your LOGO!. The cursor keys are also available as inputs as are the LOGO! TD function keys if you have a LOGO! TD module.You can also select a fixed logical level for the input.

To change an input in your LAD circuit diagram, double-click on the corresponding block to open a pop-up window in which you can make your changes.

5.1.3.2 Make contact

Make contacts, as well as break contacts and analog contacts represent the input terminals of a LOGO! module.

A pop-up window opens when you place the contact into your circuit diagram. In this dialog, you can specify the type of input according to the LOGO! used. The cursor keys are also available as inputs as are the LOGO! TD function keys if you have a LOGO! TD module. You can also select a fixed logical level for the input.

To change an input in your LAD circuit diagram, double-click on the corresponding block to open the relevant pop-up dialog.

5.1.3.3 Analog contacts

Analog contacts, as well as break contacts and make contacts represent the input terminals of a LOGO! device.

A pop-up window opens when you insert the contact in your circuit diagram. In this dialog you can specify the type of input according to your LOGO! device.

To change an input in your LAD circuit diagram, double-click on the corresponding block in your circuit diagram to open a pop-up window in which you can make your changes.

5.1.3.4 Relay coil

Relay coils represent the output terminals like inverted outputs and analog outputs on a LOGO!.

To change an output in your LAD circuit diagram, double-click on the corresponding block to open a pop-up window in which you can assign various functions to the output.

5.1.3.5 Inverted output

Inverted outputs, as well as relay coils and analog outputs represent the output terminals of a LOGO! device.

To change an output in your LAD circuit diagram, double-click on the corresponding block to open a pop-up window in which you can assign various functions to the output.

5.1.3.6 Analog output

Analog outputs represent the output terminals like relay coils and inverted output on a LOGO!.

To change an output in your LAD circuit diagram, double-click on the corresponding block to open a pop-up window in which you can assign various functions to the output.

If you connect a special function that has an analog output to a **real** analog output, then note that the analog output can only process values from 0 to 1000.

5.1.3.7 Internal flag

You can use an internal flag to terminate a current path and continue it in a new path.

Contrary to the flag block, this does not use a flag resource of your LOGO! device.

5.2 Basic functions (only FBD Editor)

5.2.1 Basic functions (FBD Editor only) - Overview

This tool has to be selected if you want to place standard Boolean logic blocks on the programming interface. The specific type of block is selected from this group from an additional toolbar that is opened when you select the **basic functions** tool.


AND


OR


AND with edge evalation


NOR


NAND


XOR


NAND with edge evaluation


NOT

Inverting the inputs

You can invert individual inputs:

- A logical "1" at a specific input is inverted to logical "0" in the circuit program.
- A logical "0" is inverted to logical "1" in the circuit program.

To do so, right-click on the input and select the **invert** command from the shortcut menu.

You cannot invert the inputs of output blocks.

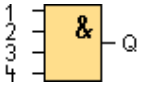
0BA0-0BA3:
To invert an input, use the basic function NOT.

Timing diagrams

Each timing diagram of the basic functions displays three inputs to make evaluation easier for you.

0BA0-0BA3:
The basic functions have three inputs.

5.2.2 AND



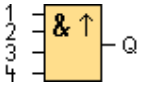
The output of an AND function is only 1 if **all** inputs are 1, that is, when they are closed.

A block input that is not used (x) is assigned: x = 1.

AND function logic table

Input 1	Input 2	Input 3	Input 4	Output
0	0	0	0	0
0	0	0	1	0
0	0	1	0	0
0	0	1	1	0
0	1	0	0	0
0	1	0	1	0
0	1	1	0	0
0	1	1	1	0
1	0	0	0	0
1	0	0	1	0
1	0	1	0	0
1	0	1	1	0
1	1	0	0	0
1	1	0	1	0
1	1	1	0	0
1	1	1	1	1

5.2.3 AND with edge evaluation

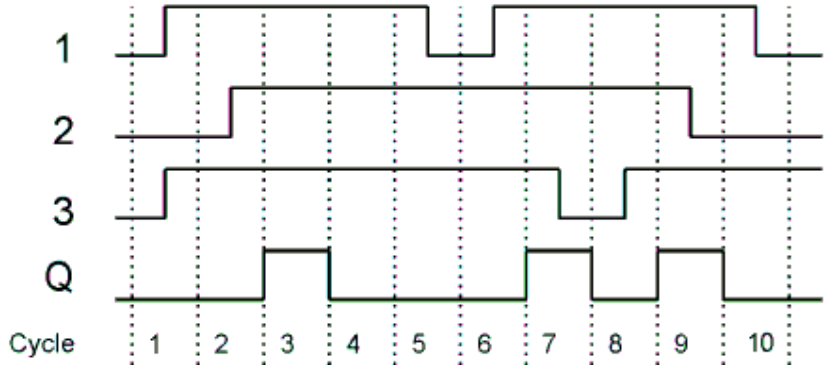


The output of an AND with edge evaluation is only 1 if **all** inputs are 1 and **at least one** input was 0 during the last cycle.

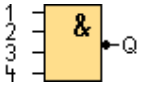
The output is set to 1 for the duration of one cycle and must be reset to 0 for the duration of the next cycle before it can be set to 1 again.

A block input that is not used (x) is assigned: x = 1.

Timing diagram of an AND with edge evaluation:



5.2.4 NAND



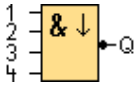
The output of an NAND function is only 0 if **all** inputs are 1, i.e. when they are closed.

A block input that is not used (x) is assigned: x = 1.

NAND function logic table

Input 1	Input 2	Input 3	Input 4	Output
0	0	0	0	1
0	0	0	1	1
0	0	1	0	1
0	0	1	1	1
0	1	0	0	1
0	1	0	1	1
0	1	1	0	1
0	1	1	1	1
1	0	0	0	1
1	0	0	1	1
1	0	1	0	1
1	0	1	1	1
1	1	0	0	1
1	1	0	1	1
1	1	1	0	1
1	1	1	1	0

5.2.5 NAND with edge evaluation

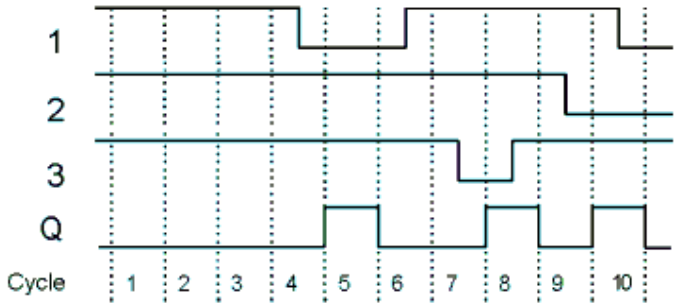


The output of a NAND with edge evaluation is only 1 if **at least one** input is 0 and **all** inputs were 1 during the last cycle.

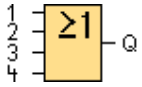
The output is set to 1 for the duration of one cycle and must be reset to 0 at least for the duration of the next cycle before it can be set to 1 again.

A block input that is not used (x) is assigned: x = 1.

Timing diagram of a NAND with edge evaluation



5.2.6 OR



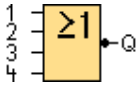
The output of an OR is 1 if **at least one** input is 1 (closed).

A block input that is not used (x) is assigned: x = 0.

OR function logic table

Input 1	Input 2	Input 3	Input 4	Output
0	0	0	0	0
0	0	0	1	1
0	0	1	0	1
0	0	1	1	1
0	1	0	0	1
0	1	0	1	1
0	1	1	0	1
0	1	1	1	1
1	0	0	0	1
1	0	0	1	1
1	0	1	0	1
1	0	1	1	1
1	1	0	0	1
1	1	0	1	1
1	1	1	0	1
1	1	1	1	1

5.2.7 NOR



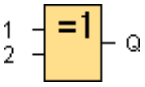
The output of a NOR (NOT OR) is only 1 if **all** inputs are 0 (open). When one of the inputs is switched on (logical 1 state), the output is switched off.

A block input that is not used (x) is assigned: x = 0.

NOR function logic table

Input 1	Input 2	Input 3	Input 4	Output
0	0	0	0	1
0	0	0	1	0
0	0	1	0	0
0	0	1	1	0
0	1	0	0	0
0	1	0	1	0
0	1	1	0	0
0	1	1	1	0
1	0	0	0	0
1	0	0	1	0
1	0	1	0	0
1	0	1	1	0
1	1	0	0	0
1	1	0	1	0
1	1	1	0	0
1	1	1	1	0

5.2.8 XOR



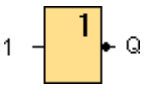
The XOR (exclusive OR) output is 1 if the signal status of the inputs is **different**.

A block input that is not used (x) is assigned: x = 0.

XOR function logic table

Input 1	Input 2	Output
0	0	0
0	1	1
1	0	1
1	1	0

5.2.9 NOT



The output is 1 if the input is 0. The NOT block inverts the input status.

Advantage of the NOT, for example: LOGO! no longer requires break contacts. You simply use a make contact and convert it into a break contact with the help of the NOT function.

NOT function logic table

Input 1	Output
0	1
1	0

5.3 Special functions

5.3.1 Special functions - Overview

This tool has to be selected if you want to place additional retentive or time-related function blocks on the programming interface. The specific type of block is selected from an additional toolbar that opens when you select the **SFB** tool.

This topic also describes the following tasks or characteristics of the special functions:

- Inverting inputs
- Block configuration
- Reference parameters
- Protection
- Retentivity

The special functions are organized by group and are listed below:

Timers

 On-delay

 Off-delay

 On-/Off-delay

 Retentive on-delay

 Wiping relay (pulse output)

 Edge triggered wiping relay

 Symmetrical pulse generator

 Asynchronous pulse generator

 Random generator

 Stairway light switch

 Multiple function switch

 Weekly timer

Yearly timer

Analog

 Analog threshold trigger

 Differential analog trigger

 Analog comparator

 Analog value monitoring

 Analog amplifier

 Analog multiplexer

 Pulse width modulator (PWM)

 Analog math

Control and Regulate

 PI controller

 Ramp control

Counters


Up/Down counter


Hours counter


Threshold trigger

Others


Latching relay


Pulse relay


Message text


Softkey


Shift register


Analog math error detection

The **LAD Editor** offers you the following additional functions:

AND with edge evaluation

NAND with edge evaluation

FBD editor: description of the blocks of special functions

The description of the blocks of special functions in the circuit diagram begins with timer blocks (“T”), with counter blocks (“C”) and with the remaining blocks (“SF”).

The LOGO! version you have selected determines these characteristics of your circuit program:

- Available blocks
- Parameters that you can set

Inverting inputs

You can invert individual inputs:

- A logical "1" at a specific input is inverted to logical "0" in the circuit program.
- A logical "0" is inverted to logical "1" in the circuit program.

To do so, right-click on the input and select the **invert** command from the shortcut menu.

You cannot invert the inputs of output blocks.

0BA0-0BA3:

To invert an input, use the basic function NOT.

Block configuration

The block properties dialog provides you with an easy means of setting the various block parameters.

Reference functionality

You can also assign parameters to blocks by means of other blocks. Such parameters are called reference parameters.

If you click on the Reference button next to a parameter in the block properties window, you can select which other block provides the actual value for that parameter. For example, if you click the Reference button for the Off-Delay parameter of an Off-Delay timer, you can then choose a specific block to use to provide the time value of the timer. LOGO!Soft Comfort displays the set of available blocks in your circuit program that you can use to provide the reference parameter.

This way it is possible, for example, to assign the time of an off-delay timer from an analog output value from another block.

A quick way of changing block parameters

Protection

If a **Protection Active** check box exists for the protection of a block parameter, you can enable or lock the display and editing of this parameter in LOGO! configuration mode.

Retentivity

The switching state and counter values of SFBs can be retentive. This means that the current data values are retained, for example after a power failure, so that the function is resumed at the break position after power on. Hence, a timer is not reset, but instead the time-to-go expires.

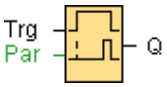
However, to enable this feature for the relevant function, retentivity needs to be set. There are two possible settings:

- on: Current data values are retained
- off: Current data values are not retained (default).

The hours counter is an exception, because it is generally retentive.

5.3.2 Timers

5.3.2.1 On-delay



Short description

The output is not switched on until a configured delay time has expired.

Connection	Description
Trg input	The on-delay time is triggered via the Trg (Trigger) input.
Parameter	**T** represents the on-delay time after which the output is switched on (output signal transition 0 to 1). **Retentivity** on = the status is retentive in memory.
Output **Q**	Q switches on after a specified time T has expired, provided Trg is still set.

Parameter T

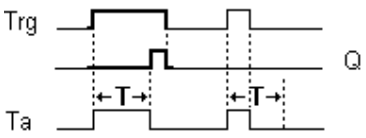
The time in parameter T can be provided by the value of another already-programmed function:

- Analog comparator: Ax - Ay
- Analog trigger: Ax
- Analog amplifier: Ax
- Analog multiplexer: AQ
- Analog ramp: AQ
- Analog math: AQ
- PI controller: AQ
- Up/Down counter: Cnt

You select the required function by the block number.

For information on the validity and accuracy of the time base, refer to the LOGO! manual.

Timing diagram



Description of the function

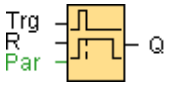
The time Ta (the current time in LOGO!) is triggered with the 0 to 1 transition at input Trg.

If the status at input Trg stays 1 at least for the duration of the configured time T, the output is set to 1 when this time has expired (the on signal of the output follows the on signal of the input with delay).

The time is reset if the status at input Trg changes to 0 again before the time T has expired.

The output is reset to 0 when input Trg is 0.

5.3.2.2 Off-delay



Short description

The output with off delay is not reset until a defined time has expired.

Connection	Description
Input **Trg**	Start the off-delay time with a negative edge (1 to 0 transition) at input Trg (Trigger).
Input **R**	Reset the off-delay time and set the output to 0 via the R (Reset) input. Reset has priority over Trg.
Parameter	**T:** The output is switched off on expiration of the delay time T (output signal transition 1 to 0). **Retentivity** on = the status is retentive in memory.
Output **Q**	Q is switched on for the duration of the time T after a trigger at input Trg.

Parameter T

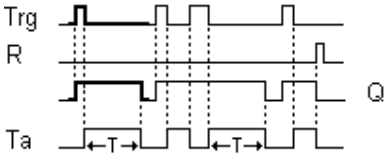
The time set in parameter T can be supplied by the value of another already-programmed function:

- Analog comparator: Ax - Ay
- Analog trigger: Ax
- Analog amplifier: Ax
- Analog multiplexer: AQ
- Analog ramp: AQ
- Analog math: AQ
- PI controller: AQ
- Up/Down counter: Cnt

You select the required function by the block number.

For information on the validity and accuracy of the time base, refer to the LOGO! manual.

Timing diagram



Description of the function

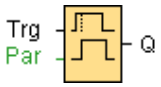
Output Q is set to 1 instantaneously with a 0 to 1 transition at input Trg.

At the 1 to 0 transition at input Trg, LOGO! retriggers the current time T, and the output remains set. The output Q is reset to 0 when T_a reaches the value specified in T (T_a=T) (off delay).

A one-shot at input Trg retriggers the time Ta.

You can reset the time Ta and the output via the input R (Reset) before the time Ta has expired.

5.3.2.3 On-/Off-delay



Short description

The on/off delay function block is used to set an output after a configured on-delay time and then reset it again upon expiration of a second configured time.

Connection	Description
input **Trg**	You trigger the on-delay with a positive edge (0 to 1 transition) at input Trg (Trigger). You trigger the off-delay with a negative edge (1 to 0 transition).
Parameter	**T_H** is the on-delay time for the output (output signal transition 0 to 1). **T_L** is the off-delay time for the output (output signal transition 1 to 0). **Retentivity** on = the status is retentive in memory.
Output **Q**	Q is switched on upon expiration of a configured time T_H if Trg is still set. It is switched off again upon expiration of the time T_L and if Trg has not been set again.

Parameters T_H and T_L

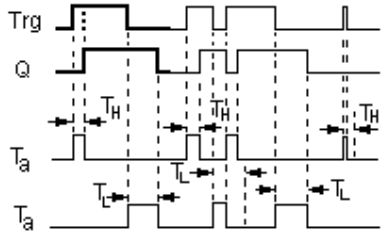
The on-delay time and off-delay time set in parameter T_H and T_L can be provided by the actual value of another already-programmed function:

- Analog comparator: Ax – Ay
- Analog trigger: Ax
- Analog amplifier: Ax
- Analog multiplexer: AQ
- Analog ramp: AQ
- Analog math: AQ
- PI controller: AQ
- Up/Down counter: Cnt

You select the required function by the block number.

For information on the validity and accuracy of the time base, refer to the LOGO! manual.

Timing diagram



Description of the function

The time T_H is triggered with a 0 to 1 transition at input Trg.

If the status at input Trg is 1 for at least the duration of the configured time T_H, the output is set to logical 1 upon expiration of this time (output is on-delayed to the input signal).

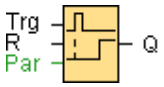
The time T_H is reset if the status at input Trg is reset to 0 before this time has expired.

The time T_L is triggered with the 1 to 0 transition at the output.

If the status at input Trg remains 0 at least for the duration of a configured time T_L, the output is reset to 0 upon expiration of this time (output is off delayed to the input signal).

The time T_L is reset if the status at input Trg returns to 1 before this time has expired.

5.3.2.4 Retentive on-delay



Short description

A one-shot at the input triggers a configurable time. The output is set upon expiration of this time.

Connection	Description
Input **Trg**	Trigger the on-delay time via the Trg (Trigger) input.
Input **R**	Reset the on-delay time and reset the output to 0 via input R (Reset). Reset takes priority over Trg.
Parameter	**T** is the on-delay time for the output (output signal transition 0 to 1). **Retentivity** on = the status is retentive in memory.
Output **Q**	Q is switched on upon expiration of the time T.

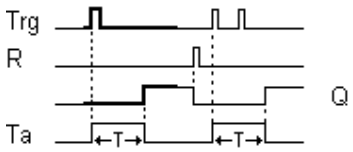
Parameter T

The time in parameter T can be provided by the value of another already-programmed function:

- Analog comparator: Ax – Ay
- Analog trigger: Ax
- Analog amplifier: Ax
- Analog multiplexer: AQ
- Analog ramp: AQ
- Analog math: AQ
- PI controller: AQ
- Up/Down counter: Cnt

You select the required function by the block number.

Timing diagram



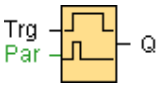
Description of the function

The current time Ta is triggered with a 0 to 1 signal transition at input Trg. Output Q is set to 1 when Ta reaches the time T. A further pulse at input Trg does not affect Ta.

The output and the time Ta are only reset to 0 with a 1 signal at input R.

If retentivity is not set, output Q and the expired time are reset after a power failure.

5.3.2.5 Wiping relay (pulse output)



Short description

An input signal generates an output signal of a configurable length.

Connection	Description
Input **Trg**	You trigger the time for the wiping relay with a signal at input Trg (Trigger).
Parameter	**T** represents the time after which the output is reset (output signal transition 1 to 0). **Retentivity** set (on) = the status is retentive in memory.
Output **Q**	A pulse at Trg sets Q. The output stays set until the time T has expired and if Trg = 1 for the duration of this time. A 1 to 0 transition at Trg prior to the expiration of T also resets the output to 0.

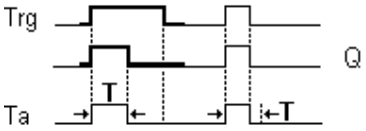
Parameter T

The off time T can be provided by the actual value of another already-programmed function:

- Analog comparator: Ax – Ay
- Analog trigger: Ax
- Analog amplifier: Ax
- Analog multiplexer: AQ
- Analog ramp: AQ
- Analog math: AQ
- PI controller: AQ
- Up/Down counter: Cnt

You select the required function by the block number.

Timing diagram



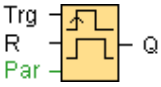
Description of the function

With the input signal Trg = 1, output Q is set to 1. The signal also triggers the time Ta, while the output remains set.

When Ta reaches the value defined at T (Ta=T), the output Q is reset to 0 state (pulse output).

If the signal at input Trg changes from 1 to 0 before this time has expired, the output is immediately reset from 1 to 0.

5.3.2.6 Edge triggered wiping relay



Short description

An input pulse generates a preset number of output pulses with a defined pulse/pause ratio (retriggerable), after a configured delay time has expired.

Connection	Description
Input **Trg**	You trigger the times for the Edge-triggered wiping relay with a signal at input Trg (Trigger).
Input **R**	The output and the current time Ta are reset to 0 with a signal at input R.
Parameter	**TH, TL:** The pulse width TH and the interpulse width TL are adjustable. **N** determines the number of pulse/pause cycles TL / TH: Value range: 1...9. **Retentivity** set (on) = the status is retentive in memory.
Output **Q**	Output Q is set when the time T_L has expired and is reset when T_H has expired.

0BA2, 0BA3:

Only the parameter T_H exists. T_H represents the off-delay time for the output.

Input R is not available

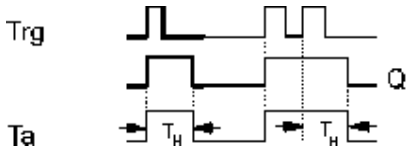
Parameters TH and TL

The pulse width TH and the interpulse width TL can be provided by the actual value of another already-programmed function:

- Analog comparator: Ax – Ay
- Analog trigger: Ax
- Analog amplifier: Ax
- Analog multiplexer: AQ
- Analog ramp: AQ
- Analog math: AQ
- PI controller: AQ
- Up/Down counter: Cnt

You select the required function by the block number.

Timing diagram



T_L = 0; N = 1

Description of the function

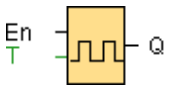
With the change at input Trg to 1, the time T_L (time low) is triggered. After the time T_L has expired, output Q is set to 1 for the duration of the time T_H (time high).

If input Trg is retriggered prior to the expiration of the preset time (T_L + T_H), the time Ta is reset and the pulse/pause period is restarted.

5.3.2.7 Symmetrical pulse generator

The symmetrical pulse generator is only available for devices of the series 0BA3.

LOGO! devices of the current series use an asynchronous pulse generator instead of the symmetrical pulse generator.

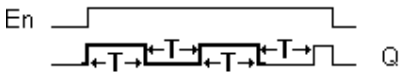


Short description

The function outputs a pulse signal with a configurable period.

Connection	Description
Input **En**	You enable (En=1) or disable (En=0) the pulse generator with the signal at input En (Enable).
Parameter	**T** is the on or off time of the output.
Output **Q**	Q is toggled periodically on and off with the pulse time T.

Timing diagram

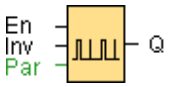


Description of function

You define the length of the on and off times at the parameter T. The **En** input enables the pulse generator. The pulse generator sets the output to 1 for the duration of the time T, then to 0 for the duration of the time T and so forth, until input En = 0.

Always specify a time T of 0.1 s. A time T is not defined for T = 0.05 s and T = 0.00 s.

5.3.2.8 Asynchronous pulse generator



Description of function

The pulse shape at the output can be modified by a configurable pulse/pause ratio.

Connection	Description
Input **En**	You enable/disable the asynchronous pulse generator with the signal at input En.
Input **Inv**	The Inv input can be used to invert the output signal of the active asynchronous pulse generator.
Parameter	**TH,TL:** You can customize the pulse width (TH) and the interpulse width (TL). **Retentivity** set (on) = the status is retentive in memory.
Output **Q**	Q is toggled on and off cyclically with the pulse/pause times T_H and T_L .

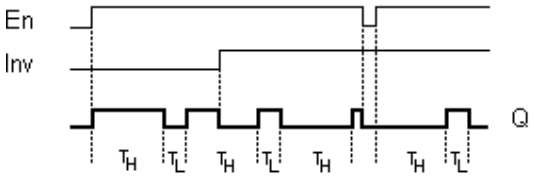
Parameters TH and TL

The pulse width TH and the interpulse width TL can be provided by the actual value of another already-programmed function:

- Analog comparator: Ax – Ay
- Analog trigger: Ax
- Analog amplifier: Ax
- Analog multiplexer: AQ
- Analog ramp: AQ
- Analog math: AQ
- PI controller: AQ
- Up/Down counter: Cnt

You select the required function by the block number.

Timing diagram

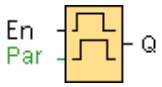


Description of the function

You can set the pulse/pause ratio at the TH (Time High) and TL (Time Low) parameters.

The INV input can be used to invert the output signal. The input block INV only inverts the output signal if the block is enabled with EN.

5.3.2.9 Random generator



Short description

The output of a random generator is toggled within a configurable time.

Connection	Description
Input **En**	The positive edge (0 to 1 transition) at the enable input En (Enable) triggers the on-delay for the random generator. The negative edge (1 to 0 transition) triggers the off-delay for the random generator.
Parameter	**TH:** The on-delay is determined at random and lies between 0 s and T_H. **TL:** The off-delay is determined at random and lies between 0 s and T_L.
Output **Q**	Q is set on expiration of the on-delay if En is still set. It is reset when the off-delay time has expired and if En has not been set again.

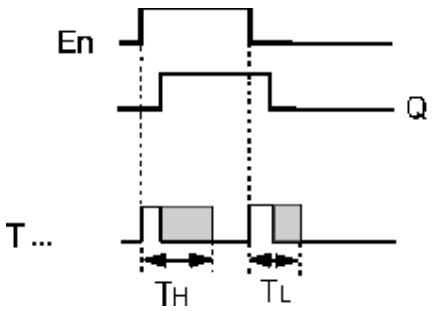
Parameters TH and TL

The on-delay time TH and the off-delay time TL can be provided by the actual value of another already-programmed function:

- Analog comparator: Ax – Ay
- Analog trigger: Ax
- Analog amplifier: Ax
- Analog multiplexer: AQ
- Analog ramp: AQ
- Analog math: AQ
- PI controller: AQ
- Up/Down counter: Cnt

You select the required function by the block number.

Timing diagram



Description of the function

With the 0 to 1 transition at input En, a random time (on-delay time) between 0 s and T_H is set and triggered. If the status at input En is 1 for at least the duration of the on-delay, the output is set to 1 when this on-delay time has expired.

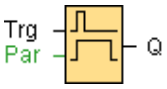
The time is reset if the status at input En is reset to 0 before the on-delay time has expired.

When input En is reset 0, a random time (off-delay time) between 0 s and T_L is set and triggered.

If the status at input En is 0 at least for the duration of the off-delay time, the output Q is reset to 0 when the off-delay time has expired.

The time is reset if the status at input En returns to 1 before the on-delay time has expired.

5.3.2.10 Stairway lighting switch



Short description

The edge of an input pulse triggers a configurable time. The output is reset when this time has expired. An off warning can be output prior to the expiration of this time.

Connection	Description
Input **Trg**	You trigger the time (off-delay) for the stairway switch with a signal at input Trg (Trigger).
Parameter	**T:** The output is reset (1 to 0 transition) when the off-delay time T has expired. **$T_!$** determines the triggering time for the prewarning. **$T_{!L}$** determines the length of the prewarning time. **Retentivity** set (on) = the status is retentive in memory.
Output **Q**	Q is reset after the time T has expired. A warning signal can be output before this time has expired.

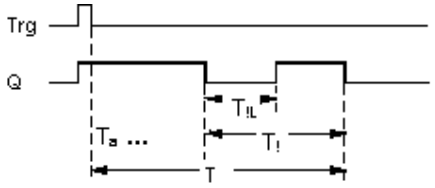
Parameters T, $T_!$ and $T_{!L}$

The off-delay time T, the prewarning time $T_!$ and the prewarning period $T_{!L}$ can be provided by the actual value of another already-programmed function:

- Analog comparator: Ax – Ay
- Analog trigger: Ax
- Analog amplifier: Ax
- Analog multiplexer: AQ
- Analog ramp: AQ
- Analog math: AQ
- PI controller: AQ
- Up/Down counter: Cnt

You select the required function by the block number.

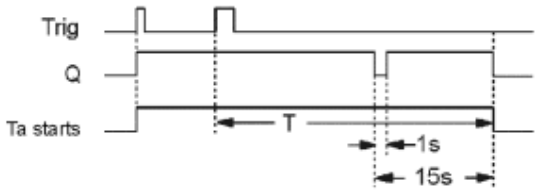
Timing diagram



0BA0 : 0BA5

0BA1, 0BA3:

The prewarning time is set to 15 s. The $T_!$ and $T_{!L}$ parameters thus become obsolete.



Changing the time base

You can change the prewarning time base and the period.

Time base T	Prewarning time	Prewarning period
Seconds*	750 ms	50 ms
Minutes	15 s	1 s
Hours	15 min	1 min

* makes sense only for programs with a cycle time of < 25 ms

Description of the function

Output Q is set to 1 with a 0 to 1 signal transition at input Trg. The 1 to 0 transition at input Trg triggers the current time and output Q remains set.

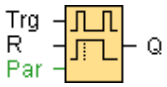
Output Q is reset to 0 when Ta reaches the time T. Before the off delay time (T - T_I) has expired, you can output a prewarning that resets Q for the duration of the off prewarning time T_{IL}.

Ta is retriggered (optional) at the next high/low transition at input Trg and if Ta is expiring.

Scan cycle time

For information on how to determine the scan cycle time of a LOGO!, refer to the appendix of the LOGO! manual.

5.3.2.11 Multiple function switch



Short description

Switch with two different functions:

- Pulse switch with off delay
- Switch (continuous light)

Connection	Description
Input **Trg**	A signal at input Trg (Trigger) sets output Q (permanent light) or resets Q with an off-delay. When active, output Q can be reset with a signal at input Trg.
Input **R**	A signal at input R resets the current time Ta and resets the output.
Parameter	**T:** determines the off-delay time. The output is reset (1 to 0 transition) when the time T expires. **T_L** determines the period during which the input must be set in order to enable the permanent light function. **$T_!$** determines the on delay for the prewarning time. **$T_{!L}$** determines the length of the prewarning time period. **Retentivity** set (on) = the status is retentive in memory.
Output **Q**	Output Q is set with a signal at input Trg, and it is reset again after a configured time has expired and depending on the pulse width at input Trg, or it is reset with another signal at input Trg.

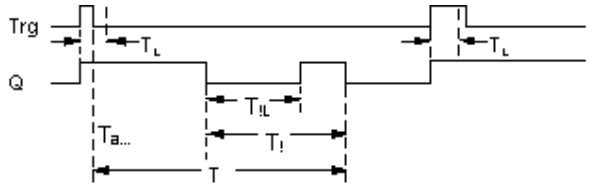
Parameters T, T_L, $T_!$ and $T_{!L}$

The off-delay time T, the permanent light time T_L, the on-delay prewarning time $T_!$, and the prewarning time period $T_{!L}$ can be provided by the actual value of another already-programmed function:

- Analog comparator: Ax – Ay
- Analog trigger: Ax
- Analog amplifier: Ax
- Analog multiplexer: AQ
- Analog ramp: AQ
- Analog math: AQ
- PI controller: AQ
- Up/Down counter: Cnt

You select the required function by the block number.

Timing diagram



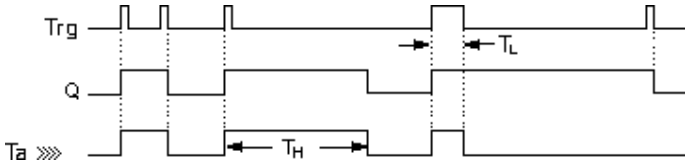
0BA2, 0BA3:

Only the T_L and T parameters exist.

The output is reset when T has expired.

T_L determines the period during which the input must be set in order to enable the permanent light function.

Input R is not available to the user.



Description of the function

Output Q is set to 1 with a 0 to 1 signal transition at Trg.

If output Q = 0, and input Trg is set hi for at least the duration of T_L, the permanent lighting function is enabled and output Q is set accordingly.

The off-delay time T is triggered when the status at input Trg changes to 0 before the time T_L has expired.

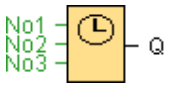
Output Q is reset when the Ta = T.

You can output an off-warning signal prior to the expiration of the off-delay time (T - $T_!$) that resets Q for the duration of the off prewarning time T_{IL}. A subsequent signal at input Trg always resets T and output Q.

Caution

The time base for T, $T_!$ and T_{IL} must be identical.

5.3.2.12 Weekly timer



Caution

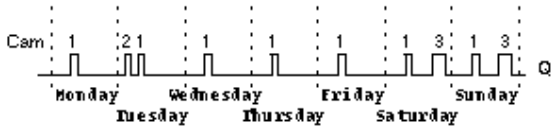
Your LOGO! must be equipped with an internal real-time clock if you are going to use this SFB.

Short description

The output is controlled by means of a configurable on/off date. The function supports any combination of weekdays.

Connection	Description
Parameters No1, No2, No3	At the **No1, No2, No3** (Cam) parameters you set the on- and off-time triggers for each cam of the weekly timer. For each Cam you specify the day of the week and the time of day for the on- and off-times.
Par	You specify whether the timer pulses on for one cycle when activated and is then reset. The pulse parameter applies to all three cams.
Output Q	Q is set when the configured cam is actuated.

Timing diagram (three practical examples)



No1:	Daily:	06:30 h to 8:00 h
No2:	Tuesday:	03:10 h to 04:15 h
No3:	Saturday and Sunday:	16:30 h to 23:10 h

Description of the function

Each weekly timer is equipped with three cams. You can configure a time hysteresis for each individual cam. At the cams you set the on- and off-hysteresis. The weekly timer sets the output at a certain time, provided it is not already set.

The weekly timer resets the output at the off-time if you configured an off-time, or at the end of the cycle if you specified a pulse output. A conflict is generated in the weekly timer when the on-time and the off-time at another cam are identical. In this case, cam 3 takes priority over cam 2, while cam 2 takes priority over cam 1.

The switching status of the weekly timer is determined by the status at the No1, No2 and No3 cams.

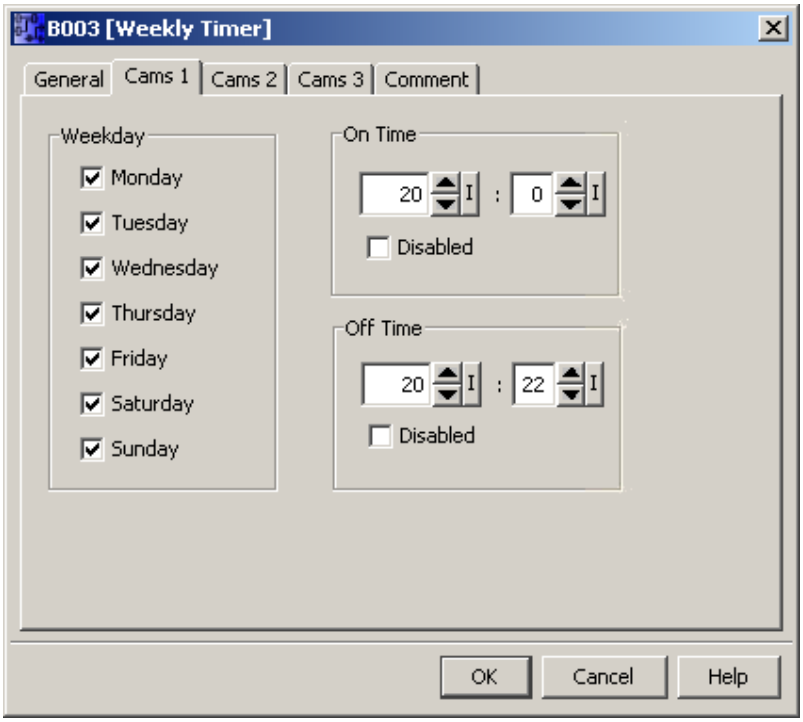
On-times

The on-time is any time between 00:00 h and 23:59 h. You can also configure the on-time to be a pulse signal. The timer block will be activated at the specified time for one cycle and then the output is reset. The off-time is disabled in this case as it is not applicable.

Special characteristics to note when configuring

The block properties window offers a tab for each one of the three cams. Here you can set the day of the week for each cam. Each tab offers you in addition an option of defining the on- and off-times for each cam in hour and minute units. Hence, the shortest switching cycle is one minute. Also on each tab you have the option of specifying a pulse output for the cam.

You can disable the on- and off-times individually. You can achieve switching cycles extending across more than one day, for example, by setting the on-time for cam 1 to Monday 7:00 h and the off-time of cam 2 to Wednesday 13:07 h, while disabling the on time for cam 2.



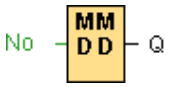
The **Pulse** setting is available only as of the 0BA6 device series.

Backup of the real-time clock

The internal real-time clock of LOGO! is buffered against power failure. The buffering time is influenced by the ambient temperature, and is typically 80 hours at an ambient temperature of 25°C.

The LOGO! 0BA6 devices and later support the option of a battery card or a combined program module (memory) and battery card. The real-time clock is buffered for several years with either of these cards.

5.3.2.13 Yearly timer



Short description

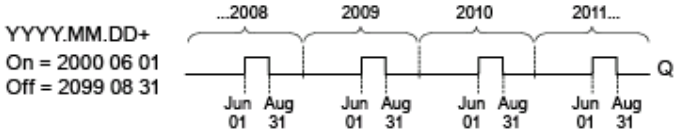
The output is controlled by means of a configurable on/off date. You can configure the timer to activate on a yearly, monthly, or user-defined time basis. With any mode, you can also configure the timer to pulse the output during the defined time period. The time period is configurable within the date range of January 1, 2000 to December 31, 2099.

Note: To use this function block, you must use a LOGO! with an internal real-time clock.

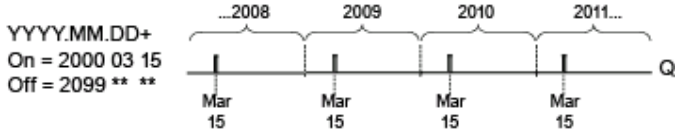
Connection	Description
Parameter	At the **No** (cam) parameter, you configure the timer mode, the on-/off-times for the timer, and whether the output is a pulse output.
Output **Q**	Q is set on when the configured cam is switched on.

Timing diagrams

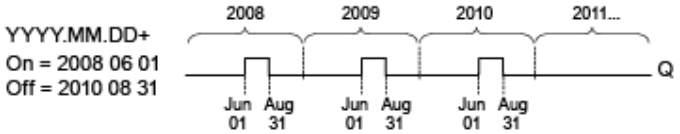
Example 1: Yearly selected, On Time = 2000.06.01, Off Time = 2099.08.31, Every year on June 1 the timer output switches on and remains on until August 31.



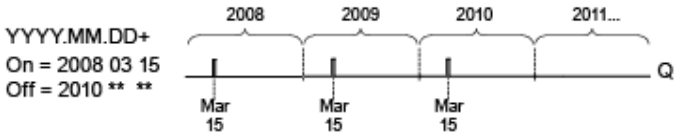
Example 2: Yearly selected, **Pulse** selected, On Time = 2000.03.15, Off Time = 2099.**.**. Every year on March 15, the timer switches on for one cycle.



Example 3: Yearly selected, On Time = 2008.06.01, Off Time = 2010.08.31. On June 1 of 2008, 2009, and 2010, the timer output switches on and remains on until August 31.

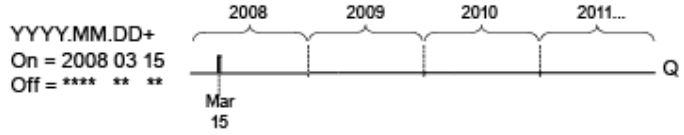


Example 4: Yearly selected, **Pulse** selected, On Time = 2008.03.15, Off Time = 2010.**.**. On March 15 of 2008, 2009, and 2010, the timer output switches on for one cycle.

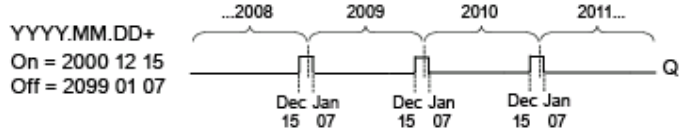


Example 5: **Monthly** not selected, **Yearly** not selected, On Time = 2008.06.01, Off Time = 2010.08.31. On June 1, 2008 the timer output switches on and remains on until August 31, 2010.

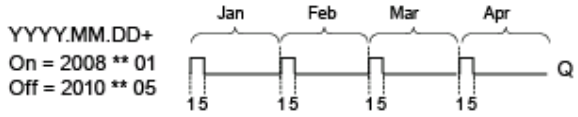
Example 6: **Monthly** not selected, **Yearly** not selected, **Pulse** selected, On Time = 2008.03.15, Off Time = ****.**.**. On March 15, 2008 the timer switches on for one cycle. Because the timer does not have a monthly action or yearly action, the timer output pulses only one time at the specified On Time.



Example 7: Yearly selected, On Time = 2000.12.15, Off Time =2010. 01.07. On December 15 of 2008 and 2009, the timer output switches on and remains on until January 7 of the following year. When the timer output turns off on January 7, 2010 it does NOT turn on again the following December 15.



Example 8: Monthly selected, On Time = 2008.**.01, Off Time = 2010.**.05. Starting in 2008, on the first day of each month the timer output switches on and switches off on the fifth day of the month. The timer continues in this pattern through the last month of 2010.



Description of the function

The yearly timer sets and resets the output at specific on and off dates. Sets and resets are executed at 00:00. If your application requires a different time, use a weekly timer together with a yearly timer in your circuit program.

The On Time specifies the month and day when the timer is set. The Off Time identifies the month and day on which the output is reset again. For the on and off times, note the order of the fields: The first field defines the year, the second the month and the third the day.

When you select the **Monthly** check box, the timer output switches on each month at the specified day of the start time and remains on until the specified day of the Off Time. The On Year specifies the initial year in which the timer is activated. The Off Year defines the last year in which the timer turns off. The maximum year is 2099.

If you select the **Yearly** check box, the timer output switches on each year at the specified month and day of the start time and remains on until the specified month and day of the Off Time. The On Year specifies the initial year in which the timer is activated. The Off Year defines the last year in which the timer turns off. The maximum year is 2099.

If you select the **Pulse** check box, the timer output switches on at the specified On Time for one cycle and then the timer output is reset. You can choose to pulse a timer on a monthly or yearly basis, or just a single time.

If you select none of the Monthly, Yearly, or Pulse check boxes, you can define a specific time period with the On Time and Off Time. It can span any time period that you choose.

For a process action that is to be switched on and off at multiple but irregular times during the year, you can define multiple yearly timers with the outputs connected by an **OR** function block.

0BA0 : 0BA5

The **Yearly** and **Pulse** settings are available only as of the 0BA6 device series

The **Monthly** setting is available only as of the 0BA4 device series.

Backup of the real-time clock

The internal real-time clock of LOGO! is buffered against power failure. The buffering time is influenced by the ambient temperature, and is typically 80 hours at an ambient temperature of 25°C. If you are using the optional LOGO! Battery card, or combined LOGO! Memory/Battery card, LOGO! can retain the clock time for up to two years.

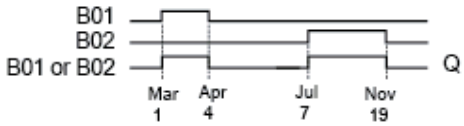
Special characteristics to note when configuring

You can numerically enter values for the month and day fields. Enter values within the logical range of month values and day values; otherwise LOGO!Soft Comfort returns an error message.

The **calendar** icon offers you an easy way of setting the date. It opens a window where you can set the days and months by clicking the relevant buttons on a calendar page.

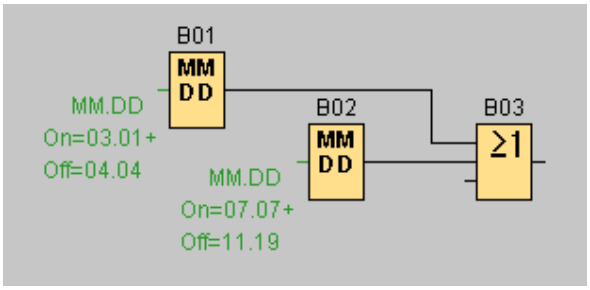
Sample configuration

The output of a LOGO! is to be switched on annually, from the 1st of March to the 4th of April and from the 7th of July to the 19th of November. This requires two blocks for configuring the specific On Times. The outputs are then linked via an OR block.



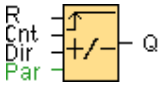
Place two yearly timer switch SFBs on your programming interface. Configure the On Time for the first yearly timer to 03.01 and the Off Time to 04.04. Configure the On Time for the second yearly timer to 07.07 and the Off Time to 11.19.

Create a logical link of the blocks with a standard OR block. The OR output is 1 if at least one of the yearly timer switches is set.



5.3.3 Counters

5.3.3.1 Up/Down counter



Short description

An input pulse increments or decrements an internal value, depending on the parameter setting. The output is set or reset when a configured threshold is reached. The direction of count can be changed with a signal at input Dir.

Connection	Description
Input **R**	You reset the output and the internal counter value to the start value (StartVal) with a signal at input R (Reset).
Input **Cnt**	This function counts the 0 to 1 transitions at input Cnt. It does not count 1 to 0 transitions. • Use the inputs I3, I4, I5, and I6 for high-frequency counts (LOGO! 12/24 RC/RCo and LOGO! 24/24o): max. 2 kHz. • Use any other input or circuit element for low-frequency counts (typically 4 Hz).
Input **Dir**	Input Dir (Direction) determines the direction of count: Dir = 0: Up Dir = 1: Down
Parameter	**On:** On threshold Value range: 0...999999 **Off:** Off threshold Value range: 0...999999 **StartVal:** Initial value from which to begin counting either down or up. **Retentivity** set (on) = the status is retentive in memory.
Output **Q**	Q is set and reset according to the actual value at Cnt and the set thresholds.

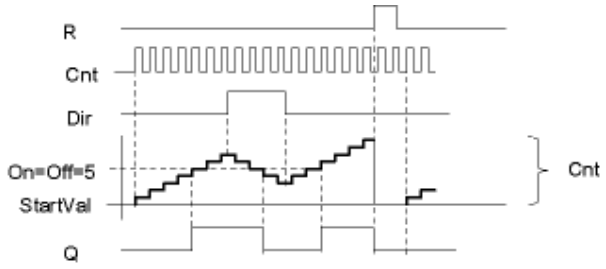
Parameters On and Off

The on threshold On and the off threshold Off can be provided by the value of another already-programmed function:

- Analog comparator: Ax – Ay
- Analog trigger: Ax
- Analog amplifier: Ax
- Analog multiplexer: AQ
- Analog ramp: AQ
- Analog math: AQ
- PI controller: AQ
- Up/Down counter: Cnt

You select the required function by the block number.

Timing diagram



Description of the function

The function increments (Dir = 0) or decrements (Dir = 1) the internal counter by one count with every positive edge at input Cnt.

You can reset the internal counter value to the start value with a signal at the reset input R. As long as R=1, the output Q is 0 and the pulses at input Cnt are not counted.

Output Q is set and reset according to the actual value at Cnt and the set thresholds. See the following rules for calculation.

Calculation rule

- If the on threshold >= off threshold, then:
  Q = 1, if Cnt >= On
  Q = 0, if Cnt < Off.
- If the on threshold < off threshold, then:
  Q = 1, if On <= Cnt < Off.

0BA0-0BA5:

The Start Value parameter does not exist. The counter always counts up or down from 0.

0BA0-0BA3:

The off parameter does not exist. The calculation rule is therefore void.

Caution

The function polls the limit value of the counter once in each cycle.

Thus, if the pulses at the fast inputs I3, I4, I5, or I6 are faster than the scan cycle time, the SFB might not switch until the specified limit has been exceeded.

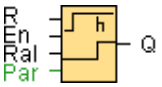
Example: Up to 100 pulses per cycle can be counted; 900 pulses have been counted so far. On = 950; Off = 10000. The output is set in the next cycle, after the value has reached 1000.

The output would not be set at all if the value Off = 980

Scan cycle time

For information on how to determine the scan cycle time of a LOGO!, refer to the appendix in the LOGO! manual.

5.3.3.2 Hours counter



Short description

A configured time is triggered with a signal at the monitoring input. The output is set when this time has expired.

Connection	Description
Input **R**	A positive edge (0 to 1 transition) at input R resets output Q and sets a configured value MI at the counter for the duration of the time-to-go (MN).
Input **En**	En is the monitoring input. LOGO! scans the On Time of this input.
Input **Ral**	A positive edge at input Ral (Reset all) resets the hours counter (OT) and the output, and sets the time-to-go value (MN) to the configured maintenance interval (MI): • Output Q = 0 • The measured operating hours OT = 0 • The time-to-go of the maintenance interval MN = MI.
Parameter	**MI**: Maintenance interval to be specified in units of hours and minutes Range of values: 0000...9999 h, 0…59 m **OT:** Accumulated total operating time. An offset start time can be specified in hours and minutes. Range of values: 00000...99999 h, 0...59 m **Q** → 0: • When "R" is selected: Q = 1, if MN = 0; Q = 0, if R = 1 or Ral = 1 • When "R+En" is selected: Q = 1, if MN = 0; Q = 0, if R = 1 or Ral = 1 or En = 0.
Output **Q**	The output is set when the time-to-go MN = 0. The output is reset: • When "Q → 0:R+En", if R = 1 or Ral = 1 or En = 0 • When "Q → 0:R", if R = 1 or Ral = 1.

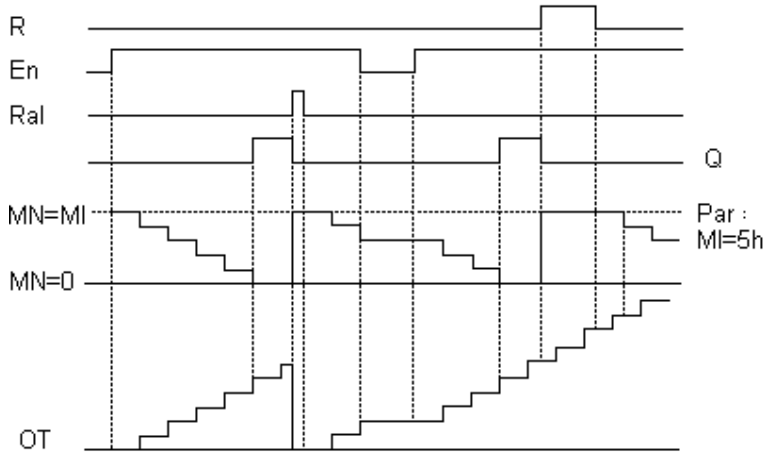
Parameter MI

The maintenance interval MI can be provided by the actual value of another already-programmed function:

- Analog comparator: Ax – Ay
- Analog trigger: Ax
- Analog amplifier: Ax
- Analog multiplexer: AQ
- Analog ramp: AQ
- Analog math: AQ
- PI controller: AQ
- Up/Down counter: Cnt

You select the required function by the block number.

Timing diagram



MI = Configured time interval

MN = Time-to-go

OT = Total time expired since the last 1 signal at the Ral input

These values are always retentive.

Description of the function

The hours counter monitors input En. As long as the status at this input is 1, LOGO! calculates the expired time and the time-to-go MN. LOGO! displays these times when set to configuration mode. The output is set to 1 when the time-to-go is equal to zero.

You reset output Q and the time-to-go counter to the specified value MI with a signal at input R. The operation hour counter OT remains unaffected.

You reset output Q and the time-to-go counter to the specified value MI with a signal at input Ral. The operation hour counter OT is reset to 0.

Depending on your configuration of the Q parameter, the output is either reset with a reset signal at input R or Ral ("Q → R"), or when the reset signal is 1 or the En signal is 0 ("Q → R+En").

Viewing the MI, MN and OT values

In LOGO!Soft Comfort you can fetch the hours counter via the Tools -> Transfer: Hours counter menu command.

Limit value of OT

The value of the operating hours in OT are retained when you reset the hours counter with a signal at input R. The hours counter OT continues the count as long as En = 1, irrespective of the status at the reset input R. The counter limit of OT is 99999 h. The hours counter stops when it reaches this value.

In programming mode, you can set the initial value of OT. The counter starts operation at any value other than zero. MN is automatically calculated at the START, based on the MI and OT values.

Example: MI = 100, OT = 130, the result is MN = 70

Parameter preset

In LOGO!Soft Comfort, you can define MI and an OT start value.

You determine that Q does not depend on En by selecting the corresponding check box.

Retentivity with the hours counter

The hours counter in the LOGO! is generally retentive.

However, if the values of the hours counter are lost after a power failure, then select the respective block in your circuit program. Right-click on the hours counter and select **Block Properties > Parameters**. The option **Retentivity** must be activated and unchangeable (grayed out).

If the **Retentivity** option is not available, then delete the block and insert a new special function **hours counter** at the same position.

0BA0-0BA5:
The maintenance interval (MI) and the start time for the operating time (OT) were in units of hours. These values could not be provided by another function prior to the 0BA6 device series.

5.3.3.3 Threshold trigger

Short description

The output is switched on and off depending on two configurable frequencies.

Connection	Description
Input Fre	The function counts 0 to 1 transitions at input Fre. Transitions from 1 to 0 are not counted. • Use the inputs I3, I4, I5, and I6 for high-frequency counts (LOGO! 12/24 RC/RCo and LOGO! 24/24o): max. 2 kHz: • Use any other input or circuit element for low frequencies (typical 4 Hz).
Parameter	**On:** On threshold Range of values: 0000...9999 **Off:** Off threshold Range of values: 0000...9999 **G_T:** Time interval or gate time during which the input pulses are measured. Range of values: 00:05 s...99:99 s
Output **Q**	Q is set or reset according to the threshold values.

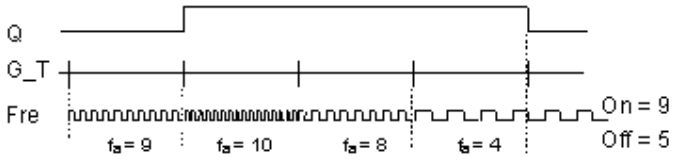
Parameter G_T

The gate time G_T can be provided by the actual value of another already-programmed function:

- Analog comparator: Ax – Ay
- Analog trigger: Ax
- Analog amplifier: Ax
- Analog multiplexer: AQ
- Analog ramp: AQ
- PI controller: AQ
- Up/Down counter: Cnt
- Analog Math AQ

You select the required function by the block number.

Timing diagram



fa = Input frequency

Description of the function

The trigger measures the signals at input Fre. The pulses are captured during a configurable period G_T.

Q is set or reset according to the set thresholds. See the following calculation rule.

Calculation rule

- If the threshold (On) > threshold (Off), then:
  Q = 1, if fa >= On
  Q = 0, if fa < Off.
- If the threshold (On) < threshold (Off), then Q = 1, if
  On <= fa < Off.

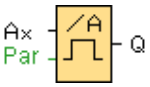
0BA0-0BA3:

The following calculation rules apply

- If Fre > threshold (On), then:
  Q = 1
- If Fre <= threshold (Off), then:
  Q = 0

5.3.4 Analog

5.3.4.1 Analog threshold trigger



Short description

The output is set or reset depending on two configurable thresholds (hysteresis).

Connection	Description
Input **Ax**	Input the analog signal to be evaluated at input Ax. Use the analog inputs AI1...AI8, the analog flags AM1...AM6, the block number of a function with analog output, or the analog outputs AQ1 and AQ2. 0 - 10 V is proportional to 0 - 1000 (internal value).
Parameter	**A:** Gain Range of values: +- 10.00 **B:** Zero offset Range of values: +- 10,000 **On:** On threshold Range of values: +- 20,000 **Off:** Off threshold Range of values: +- 20,000 **p:** Number of decimals Range of values: 0, 1, 2, 3
Output **Q**	Q is set or reset depending on the set thresholds.

Parameters On and Off

The On and Off parameters can be provided by the actual value of another already-programmed function:

- Analog comparator: Ax – Ay
- Analog trigger: Ax
- Analog amplifier: Ax
- Analog multiplexer: AQ
- Analog ramp: AQ
- Analog math: AQ
- PI controller: AQ
- Up/Down counter: Cnt

You select the required function by the block number.

0BA4:

A:	Gain
	Range of values 0.00...10.00

0BA2, 0BA3:

These parameters apply:

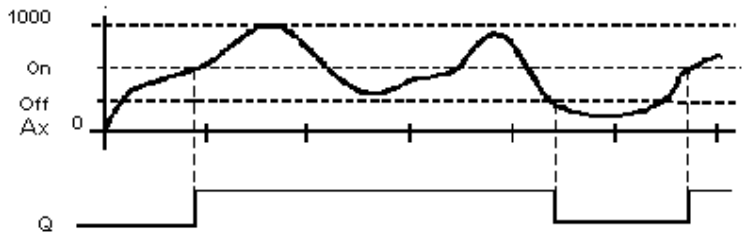
G:	Gain in [%]
	Range of values 0..1,000 %
O:	Offset
	Range of values ±999
On:	On threshold Range of values 0..9,999
Off:	Off threshold Range of values 0..9,999

Parameter p (number of decimals)

Parameter p applies only to the display of On, Off and Ax values in a message text.

Parameter p does not apply to the comparison of On and Off values. (The compare function ignores the decimal point.)

Timing diagram



Description of the function

The function reads the value of the signal at the analog input Ax.

This value is multiplied by the value of parameter A (gain). Parameter B (offset) is added to the product, hence

(Ax * Gain) + Offset = Actual value Ax.

Output Q is set or reset depending on the set threshold values. See the following calculation rule.

0BA2, 0BA3:

The **function** is as follows:

The offset parameter is added to the read analog value. The sum is multiplied by the value of the gain parameter.

Value = (AI+offset)*gain

Output Q is set to 1 if the calculated value exceeds the on threshold (TH high).

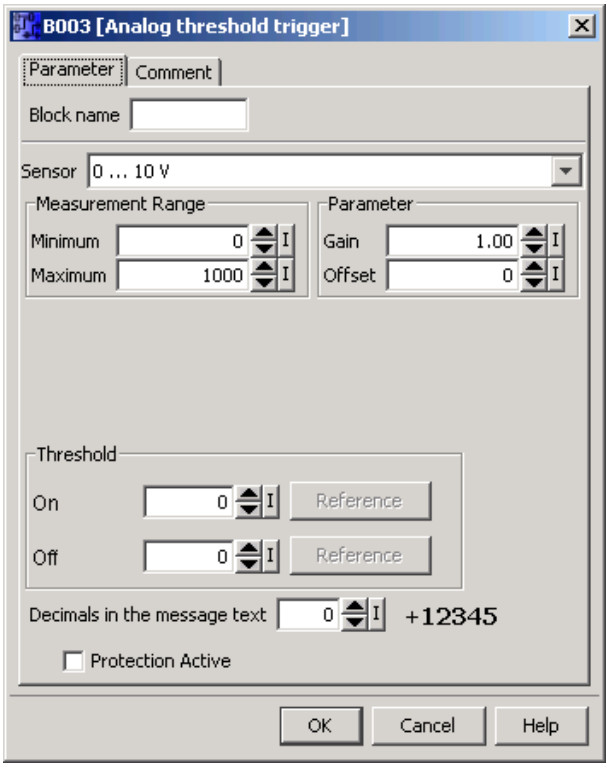
Q is reset to 0 when the value reaches or drops below the off threshold (TH low).

Calculation rule

- If threshold (On) >= threshold (Off), then:
  Q = 1, if the actual value Ax > On
  Q = 0, if the actual value Ax <= Off.
- If threshold (On) < threshold (Off), then Q = 1, if
  On <= the actual value Ax < Off.

Particular characteristics to be noted when configuring

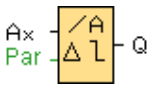
Refer to the help on analog block parameters in the Analog value processing section.



Note

The decimal point setting must be identical in the minimum and maximum range.

5.3.4.2 Analog differential trigger



Short description

The output is set and reset depending on a configurable threshold and a differential value.

Connection	Description
Input **Ax**	You apply the analog signal to be analyzed at input Ax. Use the analog inputs AI1...AI8, the analog flags AM1...AM6, the block number of a function with analog output, or the analog outputs AQ1 and AQ2. 0 - 10 V is proportional to 0 - 1000 (internal value).
Parameter	**A:** Gain Range of values: +- 10.00 **B:** Zero offset Range of values: +- 10,000 **On**: On threshold Range of values: +- 20,000 **Delta:** Differential value for calculating the off parameter Range of values: +- 20,000 **p:** Number of decimals Range of values: 0, 1, 2, 3
Output **Q**	Q is set or reset, depending on the threshold and difference values.

0BA4:

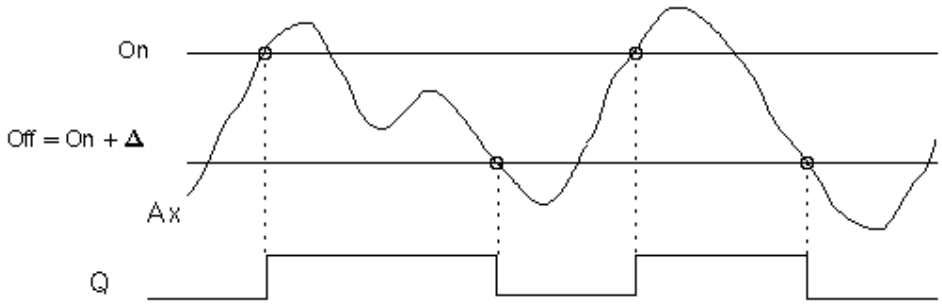
A: Gain
Range of values 0.00...10.00

Parameter p (number of decimals)

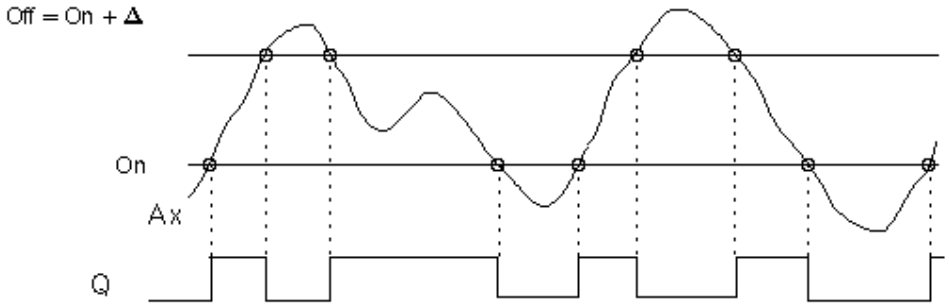
Parameter p applies only to the display of On, Off and Ax values in a message text.

Parameter p does not apply to the comparison of On and Off values. (The compare function ignores the decimal point.)

Timing diagram A: Function with negative difference Delta



Timing diagram B: Function with positive difference Delta



Description of the function

The function fetches the analog signal at input Ax.

Ax is multiplied by the value of the A (gain) parameter, and the value at parameter B (offset) is added to product, i.e.

(Ax * gain) + offset = actual value of Ax.

Output Q is set or reset, depending on the set (On) threshold and difference value (Delta). The function automatically calculates the Off parameter: Off = On + Delta, whereby Delta may be positive or negative. See the calculation rule below.

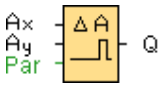
Calculation rule

- When you set a negative differential value Delta, the On threshold >= Off threshold, and:
  Q = 1, if the actual value Ax > On
  Q = 0, if the actual value Ax <= Off.
  See the timing diagram A.
- When you set a positive differential value Delta, the On threshold < the Off threshold, and Q = 1, if:
  On <= the actual value Ax < Off.
  See the timing diagram B.

Particular characteristics to be noted when configuring

Refer to the help on analog block parameters in the Information on analog value processing section.

5.3.4.3 Analog comparator



Short description

The output is set and reset depending on the difference Ax – Ay and on two configurable thresholds.

Connection	Description
Inputs **Ax, Ay**	Input the analog signals of which you want to determine the delta at the inputs Ax and Ay. Use the analog inputs AI1...AI8, the analog flags AM1...AM6, the block number of a function with analog output, or the analog outputs AQ1 and AQ2. AI1..AI8: 0 - 10 V corresponds with 0 - 1000 (internal value).
Parameter	**A:** Gain Range of values: +- 10.00 **B:** Zero offset Range of values: +- 10,000 **On**: On threshold Range of values: +- 20,000 **Off:** Off threshold Range of values: +- 20,000 **p:** Number of decimals Range of values: 0, 1, 2, 3
Output **Q**	Q is set or reset depending on the set thresholds.

0BA4:

A: Gain

Range of values 0.00...10.00

0BA0-0BA3:

The following parameters apply:

G : Gain in [%]

Range of values: 0..1000 %

O : Offset

Range of values: ±999

delta : Threshold

Q is set to 1 when the difference Ax-Ay exceeds the threshold.

Parameters On and Off

The on threshold On and the off threshold Off can be provided by the actual value of another already-programmed function:

- Analog comparator: Ax – Ay
- Analog trigger: Ax
- Analog amplifier: Ax
- Analog multiplexer: AQ
- Analog ramp: AQ
- Analog math: AQ
- PI controller: AQ
- Up/Down counter: Cnt

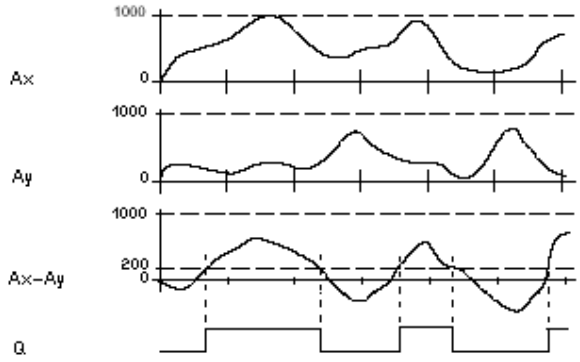
You select the required function by the block number.

Parameter p (number of decimals)

Parameter p applies only to Ax, Ay, Delta, On and Off values displayed in a message text.

Parameter p does not apply to the comparison of on and off values. (The compare function ignores the decimal point.)

Timing diagram



Q for Ax – Ay > 200, On = Off = 200

Description of the function

The function reads the value of the signal at the analog input Ax.

This value is multiplied by the value of parameter A (gain). Parameter B (offset) is added to the product, hence

(Ax * gain) + offset = Actual value Ax.

(Ay * gain) + offset = Actual value Ay.

Output Q is set or reset depending on the difference of the actual values Ax – Ay and the set thresholds. See the following calculation rule.

Calculation rule

- If threshold On >= threshold Off, then:
  Q = 1, if (actual value Ax - actual value Ay) > On
  Q = 0, if (actual value Ax - actual value Ay) <= Off.
- If threshold On < threshold Off, then Q = 1, then:
  On <= (actual value Ax - actual value Ay) < Off.

0BA2, 0BA3:

The following **functions/calculation rules** apply

The function adds the relevant specified offset to the analog values Ax and Ay. The sum is multiplied with the value of the gain parameter. The difference is formed from both calculated values.

Output Q is set if this difference between these values exceeds the threshold you have configured under delta.

Calculation rule:

Q = 1, if:

((Ax + offset) * gain) - ((Ay + offset)*gain)>threshold delta

Q is reset to 0 when the threshold reaches or drops below delta.

Reducing the input sensitivity of the analog comparator

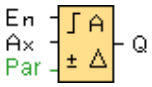
You can delay the output of the analog comparator selectively by means of the "on delay" and "off delay" SFBs. By doing so, you determine that output Q is only set if the input trigger length Trg (= output of the analog comparator) exceeds the defined on delay time.

This way you can set a virtual hysteresis, which renders the input less sensitive to short changes.

Particular characteristics to be noted when configuring

For help on analog block parameters, refer to the Analog value processing section.

5.3.4.4 Analog watchdog



Short description

This special function saves the process variable of an analog input to memory, and sets the output when the output variable exceeds or drops below this stored value plus a configurable offset.

Connection	Description
Input **En**	A positive edge (0 to 1 transition) at input En saves the analog value at input Ax ("Aen") to memory and starts monitoring of the analog range Aen +- Delta.
Input **Ax**	You apply the analog signal to be monitored at input Ax. Use the analog inputs AI1...AI8, the analog flags AM1...AM6, the block number of a function with analog output, or the analog outputs AQ1 and AQ2. 0 - 10 V is proportional to 0 - 1000 (internal value).
Parameter	**A:** Gain Range of values: +- 10.00 **B:** Zero offset Range of values: +- 10,000 **Threshold 1:** Difference value above Aen: on/off threshold Range of values: 0 – 20,000 **Threshold 2:** Difference value below Aen: on/off threshold Range of values: 0 – 20,000 **p:** Number of decimals Range of values: 0, 1, 2, 3
Output **Q**	Q is set/reset, depending on the stored analog value and the offset.

Parameters Threshold 1 and Threshold 2

The two threshold parameters Threshold 1 and Threshold 2 can be provided by the actual value of another already-programmed function:

- Analog comparator: Ax – Ay
- Analog trigger: Ax
- Analog amplifier: Ax
- Analog multiplexer: AQ
- Analog ramp: AQ
- Analog math: AQ
- PI controller: AQ
- Up/Down counter: Cnt

You select the required function by the block number.

The thresholds are represented by the symbol Δ on the LOGO! Basic module and in the timing diagram below.

0BA4:

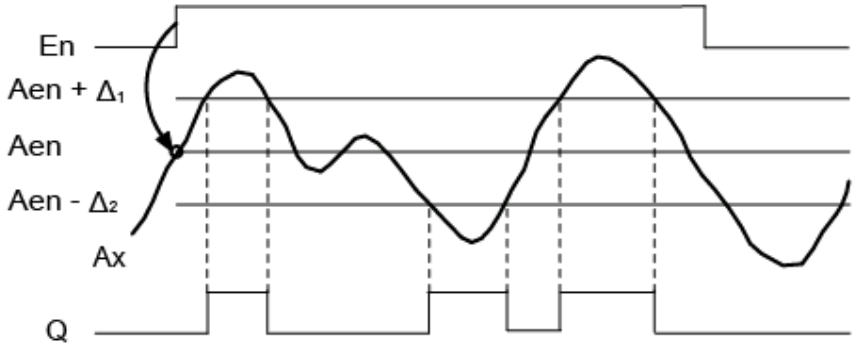
A: Gain

Range of values 0.00...10.00

Parameter p (number of decimals)

Parameter p applies only to the display of Aen, Ax, Threshold 1 and Threshold 2 values in a message text.

Timing diagram



Description of the function

A 0 to 1 transition at input En saves the value of the signal at the analog input Ax. This saved process variable is referred to as "Aen".

Both the analog actual values Ax and Aen are multiplied by the value at parameter A (gain), and parameter B (offset) is then added to the product, as follows:

(Ax * gain) + offset = Actual value Aen, when input En changes from 0 to 1, or

(Ax * gain) + offset = Actual value Ax.

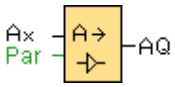
Output Q is set when the signal at input En = 1 and if the actual value at input Ax is out of range of Aen + Threshold 1 / Aen - Threshold 2.

Output Q is reset, when the actual value at input Ax lies within the range of Aen + Threshold 1 / Aen - Threshold 2, or when the signal at input En changes to lo.

Particular characteristics to be noted when configuring

Refer to the help on analog block parameters in the Analog value processing section.

5.3.4.5 Analog amplifier



Short description

This SFB amplifies an analog input value and returns it at the analog output.

Connection	Description
Input **Ax**	Input the analog signal to be amplified at input Ax. Use the analog inputs AI1...AI8, the analog flags AM1...AM6, the block number of a function with analog output, or the analog outputs AQ1 and AQ2. AI1..AI8: 0 - 10 V corresponds with 0 - 1000 (internal value).
Parameter	**A:** Gain Range of values: +- 10.00 **B:** Zero offset Range of values: +- 10000 **p:** Number of decimals Range of values: 0, 1, 2, 3
Output A**Q**	Analog output Value range for AQ: -32768...+32767

0BA4:

A: Gain
Range of values 0.00...10.00

Parameter p (number of decimals)

Parameter p applies only to the display of Ax and Ay values in a message text.

Parameter p does not apply to the comparison of On and Off values. (The compare function ignores the decimal point.)

Description of the function

The function reads the value of an analog signal at the analog input Ax.

This value is multiplied by the gain parameter A. Parameter B (offset) is added to the product, as follows:

(Ax * gain) + offset = Actual value Ax.

The actual value Ax is output at AQ.

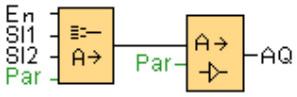
Particular characteristics to be noted when configuring

For help on analog block parameters, refer to the Analog value processing section.

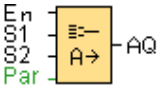
Analog output

If you connect this special function to a real analog output, then note that the analog output can only process values from 0 to 1000. To do this, connect an additional amplifier between the analog output of the special function and the real analog output. With this amplifier you standardize the output range of the special function to a value range of 0 to 1000.

Example: additional amplifier behind an analog multiplexer.



5.3.4.6 Analog multiplexer



Short description

This special function displays 0 or one of 4 saved analog values on the analog output.

Connection	Description
Input **En**	1 on input En (Enable) switches, dependent on S1 and S2, a parameterized analog value to the output AQ. 0 on input EN switches 0 to the output AQ.
Inputs **S1** and **S2**	S1 and S2 (selectors) for selecting the analog value to be issued. S1 = 0 and S2 = 0: The value 1 is issued S1 = 0 and S2 = 1: The value 2 is issued S1 = 1 and S2 = 0: The value 3 is issued S1 = 1 and S2 = 1: The value 4 is issued
Parameter	**V1...V4**: Analog values (Value) that will be issued. Value range: -32768...+32767 **p**: Number of decimal places value range: 0, 1, 2, 3
Output **AQ**	Analog output Value range for AQ: -32768...+32767

Parameters V1…V4

The values for V1…V4 can be provided by the value of another already-programmed function:

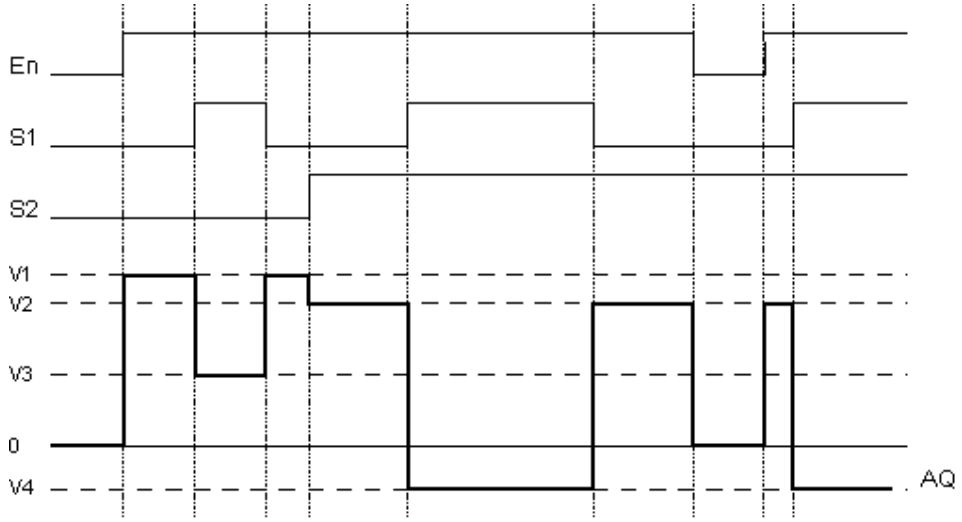
- Analog comparator: Ax – Ay
- Analog trigger: Ax
- Analog amplifier: Ax
- Analog multiplexer: AQ
- Analog ramp: AQ
- Analog math: AQ
- PI controller: AQ
- Up/Down counter: Cnt

You select the required function by the block number.

Parameter p (number of decimal places)

Parameter p applies only to the display of AQ, V1, V2, V3 and V4 values in a message text.

Timing diagram



Description of function

If input En is set, then the function issues one of 4 possible analog values V1 to V4 at the output AQ, depending on the parameters S1 and S2.

If the input En is not set, then the function issues the analog value 0 at output AQ.

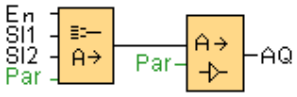
Particular characteristics to be noted when configuring

For help on analog block parameters, refer to the Analog value processing section.

Analog output

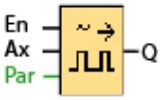
If you connect this special function to a real analog output, then note that the analog output can only process values from 0 to 1000. To do this, connect an additional amplifier between the analog output of the special function and the real analog output. With this amplifier you standardize the output range of the special function to a value range of 0 to 1000.

Example: additional amplifier behind an analog multiplexer.



Overview

5.3.4.7 Pulse Width Modulator (PWM)



Short description

The Pulse Width Modulator (PWM) instruction modulates the analog input value Ax to a pulsed digital output signal. The pulse width is proportional to the analog value Ax.

Connection	Description
Input **Ax**	Analog signal to be modulated to a pulsed digital output signal.
Parameter	**A:** Gain Range of values: +- 10.00 **B:** Zero offset Range of values: +- 10,000 **PT:** Periodic time over which the digital output is modulated **p:** Number of decimals Range of values: 0, 1, 2, 3
Output **Q**	Q is set or reset for the proportion of each time period according to the proportion of the standardized value Ax to the analog value range.

0BA1- 0BA5: The PWM function block did not exist prior to 0BA6.

Parameter PT

The periodic time PT can be provided by the actual value of another already-programmed function:

- Analog comparator: Ax – Ay
- Analog trigger: Ax
- Analog amplifier: Ax
- Analog multiplexer: AQ
- Analog ramp: AQ
- Analog math: AQ
- PI controller: AQ
- Up/Down counter: Cnt

Parameter p (number of decimals)

Parameter p applies only to the display of the Ax value in a message text.

Description of the function

The function reads the value of the signal at the analog input Ax.

This value is multiplied by the value of parameter A (gain). Parameter B (offset) is added to the product, as follows:

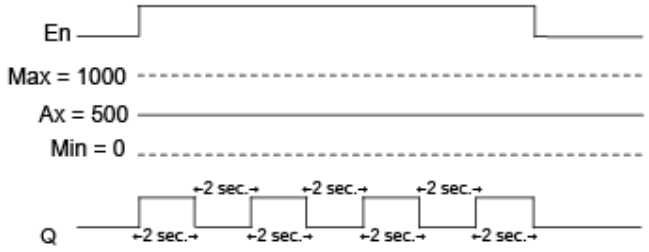
(Ax * Gain) + Offset = Actual value Ax

The function block calculates the proportion of the value Ax to the range. The block sets the digital output Q high for the same proportion of the PT (periodic time) parameter, and sets Q low for the remainder of the time period.

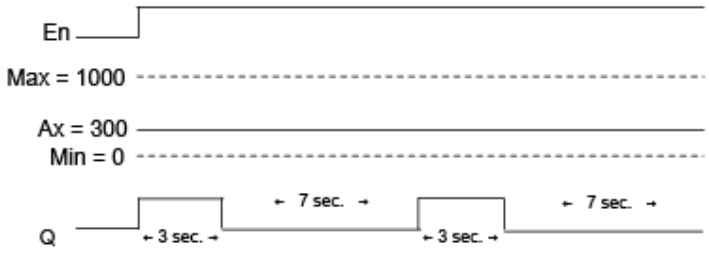
Examples with Timing Diagrams

The following examples show how the PWM instruction modulates a digital output signal from the analog input value:

1. An analog value of 500 (range 0...1000) as the value for Ax must be modulated to a digital signal string. The user-defined PT (periodic time) parameter is 4 seconds. At the digital output of the PWM function the digital signal string is 2 seconds high, 2 seconds low, 2 seconds high, 2 seconds low and continues in that pattern as long as parameter "En" = high.



2. An analog value of 300 (range 0...1000) as the value for Ax must be modulated to a digital signal string. The user-defined PT (periodic time) parameter is 10 seconds. At the digital output of the PWM function the digital signal string is 3 seconds high, 7 seconds low, 3 seconds high, 7 seconds low and continues in that pattern as long as parameter "En" = high.



Calculation rule

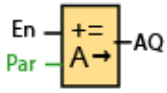
Q = 1, for (Ax – Min) / (Max – Min) of time period PT
Q = 0, for PT – [(Ax – Min) / (Max – Min)] of time period PT.

Note: Ax in this calculation refers to the actual value Ax as calculated using the Gain and Offset. Min and Max refer to the minimum and maximum values specifed for the range.

Particular characteristics to be noted when configuring

Refer to the help on analog block parameters in the Analog value processing section.

5.3.4.8 Analog math



Short description

The analog math block calculates the value AQ of an equation formed from the user-defined operands and operators.

Connection	Description
Input **En**	Enable the analog math function block.
Parameter	**V1:** Value 1: First operand **V2:** Value 2: Second operand **V3:** Value 3: Third operand **V4:** Value 4: Fourth operand **Operator1:** First operator **Operator2:** Second operator **Operator3:** Third operator **Priority1:** Priority of first operation **Priority2:** Priority of second operation **Priority3:** Priority of third operation **p:** Number of decimals Range of values: 0, 1, 2, 3
Output **AQ**	The output AQ is the result of the equation formed from the operand values and operators. AQ will be set to 32767 if a divide by 0 or overflow occurs, and -32768 if a negative overflow (underflow) occurs.

0BA1- 0BA5: The Analog Math function block did not exist prior to 0BA6.

Parameters V1, V2, V3, and V4

The values V1, V2, V3, and V4 can be provided by the actual value of another already-programmed function:

- Analog comparator: Ax – Ay
- Analog trigger: Ax
- Analog amplifier: Ax
- Analog multiplexer: AQ
- Analog ramp: AQ
- Analog math: AQ
- PI controller: AQ
- Up/Down counter: Cnt

Parameter p (number of decimals)

Parameter p applies to the display of V1, V2, V3, V4 and AQ in a message text.

Description of the function

The analog math function combines the four operands and three operators to form an equation. The operator can be any one of the four standard operators: +, -, *, or /. For each operator, you must set a unique priority of High ("H"), Medium ("M"), or Low ("L"). The high operation will be performed first, followed by the medium operation, and then by the low operation. You must have exactly one operation of each priority. The operand values can reference another previously-defined function to provide the value.

The number of operand values is fixed at four and the number of operators is fixed at 3. If you need to use fewer operands, use constructions such as " + 0" or " * 1" to fill the remaining parameters.

You can also configure the behavior of the function when the Enable parameter "En"=0. The function block can either retain its last value or be set to 0.

Possible errors: Zero division and overflow

If the analog math function block execution results in zero division or overflow, it sets internal bits that indicate the type of error that occurred. You can program an analog math error detection function block in your circuit program to detect these errors, and to control the program behavior as needed. You program one analog math error detection function block to reference one specific analog math function block.

Examples

The following tables show some simple example analog math block parameters, and the resulting equations and output values:

V1	Operator1 (Priority 1)	V2	Operator2 (Priority 2)	V3	Operator3 (Priority 3)	V4
12	+ (M)	6	/ (H)	3	- (L)	1

Equation: (12 + (6 / 3)) - 1

Result: 13

V1	Operator1 (Priority 1)	V2	Operator2 (Priority 2)	V3	Operator3 (Priority 3)	V4
2	+ (L)	3	* (M)	1	+ (H)	4

Equation: 2 + (3 * (1 + 4))

Result: 17

V1	Operator1 (Priority 1)	V2	Operator2 (Priority 2)	V3	Operator3 (Priority 3)	V4
100	- (H)	25	/ (L)	2	+ (M)	1

Equation: (100 – 25) / (2 + 1)

Result: 25

5.3.5 Analog value processing

5.3.5.1 Basics

Analog and digital

An analog signal is a physical quantity, which, within a given range, can adopt any value - any continuous intermediate value. The opposite of analog is *digital.* A digital signal knows just two states: 0 and 1 or “off” and “on”.

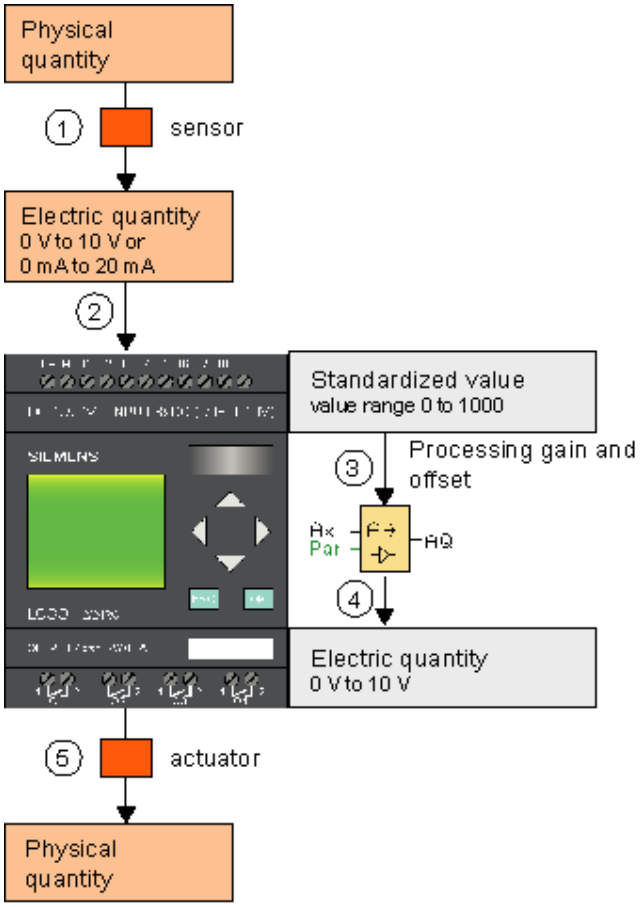
From electrical signal to analog value

Basic order of events

Several steps are required for LOGO! to process physical quantities:

1. LOGO! can read in electric voltages from 0 V to 10 V or electric currents from 0 mA to 20 mA to one analog input.
   The physical quantities (e.g. temperature, pressure, speed etc.) must therefore be converted into one electric quantity. This conversion is performed by an external sensor.
2. LOGO! reads in the electric quantity and, with further processing, this is converted into a standardized value within the range 0 to 1000. This value is then applied in the circuit program on the input of an analog special function.
3. In order to adapt the standardized value to the application, LOGO! uses an analog special function, while taking into consideration the gain and offset, to calculate the analog value. The analog value is then evaluated by the special function (e.g. analog amplifier). If an analog special function has an analog output, then the analog value is also applied to the output of the special function.
4. With the LOGO! you can also convert analog values back into an electric voltage. In doing so, the voltage can adopt values between 0 and 10 V.
5. Using this voltage, LOGO! can control an external actuator, which converts the voltage and also the analog value back into a physical quantity.

The following diagram illustrates this order of events.



Gain

The standardized value is multiplied with a parameter. Using this parameter you can more or less boost the electric quantity. Hence, this parameter is called the “gain”.

Zero point offset

You can add or subtract a parameter to or from the boosted standardized value.

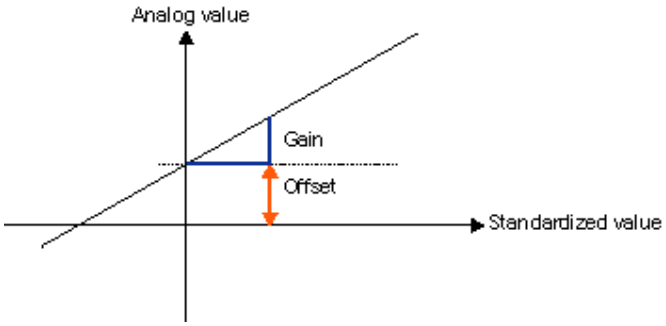
Using this parameter you can more or loss move the zero point of the electric quantity. Hence, this parameter is called the “zero point offset”.

Gain and offset

The analog value is therefore calculated as follows:

Analog value = (standardized value x gain) + offset

The following diagram illustrates this formula and the significance of gain and offset:



The straight line in the graphic describes which standardized value is being converted into which analog value. Gain corresponds to the slope of the straight line and offset to the movement of the zero passage of the straight line on the y-axis.

Analog output

If you connect a special function (that has an analog output) to a **real** analog output, then note that the analog output can only process values from 0 to 1000.

Possibile settings with LOGO!Soft Comfort

Possible settings with LOGO!

Example

Heating control

0BA0 to 0BA4

5.3.5.2 Possible settings with LOGO!Soft Comfort

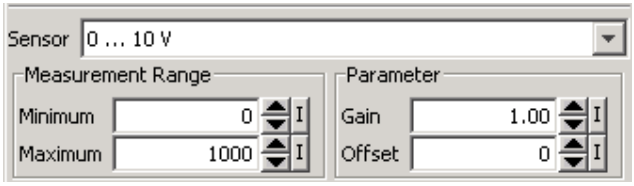
Sensor

Set your sensor type. (0 to 10 V; 0 to 20 mA; 4 to 20 mA; PT100; no sensor)

With sensor type 4 to 20 mA the value range for the standardized value is 200 and 1000.

Measurement range

Stipulate the measurement range. The measurement range is the value range shown for the analog value.



LOGO!Soft Comfort then automatically calculates the gain and offset from this.

Gain and offset

If you want to set the gain yourself, then you can enter values between -10.00 and 10.00. The value 0 makes no sense, as, irrespective of the applied analog value, you will always obtain the value 0 as a result.

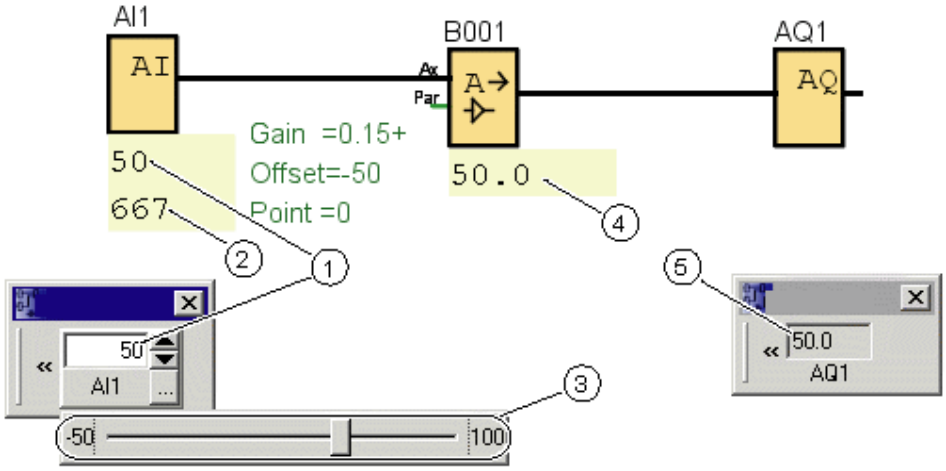
If you wish to set the offset yourself, then enter values between –10,000 and +10,000.

Rounding error

LOGO!Soft Comfort calculates the gain and zero point offset with utmost precision. However, LOGO! calculates internally using whole numerical values. Therefore, not all parameter combinations are possible on LOGO!. In this case, prior to transferring to LOGO! you can make LOGO!Soft Comfort aware of a rounding error and propose a possible replacement value range

Simulation in LOGO!Soft Comfort

With simulation in LOGO!Soft Comfort you can read the following values:



1. Physical value; the ranges are stipulated by the measuring range (3)
2. Standardized value
3. Measurement range
4. Analog value (after processing gain and offset)
5. Analog value on the analog output

5.3.5.3 Possible settings with LOGO!

If you directly program your circuit program on the LOGO!, then you can only enter the gain and offset parameters. You can calculate the gain and offset as follows:

External value range min_{Sensor} - max_{Sensor}

Range of a physical quantity that the sensor can measure.

Standardized value range min_{norm} - max_{norm}:

Value range of the standardized values.

With sensors that provide 0 to 10 V or 0 to 20 mA, the standardized value range is 0 – 1000.

With sensors that provide 4 to 20 mA, the standardized range is 200 – 1000.

So for gain and offset it follows:

Gain = $(max_{Sensor} - min_{Sensor}) / (max_{norm} – min_{norm})$

Offset = $[(min_{Sensor} \times max_{norm}) – (max_{Sensor} \times min_{norm})] / (max_{norm} – min_{norm})$

If you have calculated either the gain or the offset in accordance with the formulas above, you can then calculate the respective other value in accordance with the following formula:

Gain = $(min_{Sensor} - Offset) / min_{norm}$

Offset = $(min_{Sensor} – (Gain \times min_{norm})$

5.3.5.4 Example

Prerequisites

Sensor: temperature sensor, measuring range –50 to 100°C

Temperature to be measured 25°C

Order of events with LOGO!Soft Comfort

1. The sensor converts the temperature from 25°C to a voltage value of 5.0 V.
2. LOGO! converts the 5.0 V to the standardized value 500.
3. Using the sensor and measurement range data, LOGO! ascertains the value 0.15 for the gain and the value –50 for the offset.
   According to the formula:
   Analog value = (standardized value x gain) + offset
   LOGO! calculates as analog value:
   Analog value = (500 x 0.15) - 50 = 25

Order of events with LOGO!

1. The sensor converts the temperature from 25°C to a voltage value of 5.0 V.
2. LOGO! converts the 5.0 V to the standardized value 500.
3. From the sensor and measuring range data you must establish the values for gain and offset.
   According to the formulas:
   Gain = (maxSensor – minSensor) / (maxnorm – minnorm)
   and
   Offset = minSensor – (Gain x minnorm)
   it follows that
   Gain = (100 – (-50)) / (1000 – 0) = 0.15
   Offset = -50 – (0.15 x 0) = -50
4. According to the formula
   Analog value = (standardized value x gain) + offset
   LOGO! calculates as analog value:
   Analog value = (500 x 0.15) - 50 = 25

Additional examples

Physical quantity	Electric quantity of sensor	Standardized value	Gain	Offset	Analog value
	0 V 5 V 10 V	0 500 1000	0.01	0	0 5 10
	4 mA 12 mA 20 mA	0 500 1000	10	0	0 5000 10000
	0 mA 10 mA 20 mA	0 500 1000	1	50	50 550 1050
1000 mbar 3700 mbar 5000 mbar	0 V 6.75 V 10 V	0 675 1000	4	1000	1000 3700 5000
-30 °C 0 °C 70 °C	0 mA 6 mA 20 mA	0 300 1000	0.1	-30	-30 0 70

5.3.5.5 0BA0 to 0BA5

Restriction for device family 0BA4

The Gain cannot be a negative value.

Calculation with the device families 0BA0 to 0BA3

With LOGO! devices from these device families the parameter offset is added or subtracted to or from the standardized value **before** the value is multiplied with the parameter gain.

Hence, the following formulas apply:

Analog value = (standardized value + offset) x (gain x 100)

Gain (in percent) = $(max_{Sensor} - min_{Sensor}) / [(max_{norm} – min_{norm}) \times 100]$

Offset = $[(min_{Sensor} \times max_{norm}) – (max_{Sensor} \times min_{norm})] / (max_{Sensor} – min_{Sensor})$

Gain (in percent) = $min_{Sensor} / [(min_{norm} + offset) \times 100]$

Offset = $[max_{Sensor} / (gain \times 100)] - max_{norm}$

Gain

This parameter is given in %.

The Gain cannot be a negative value.

Zero point offset

Here you can enter values between -999 and + 999.

5.3.6 Control and regulate

5.3.6.1 Control and regulate basics

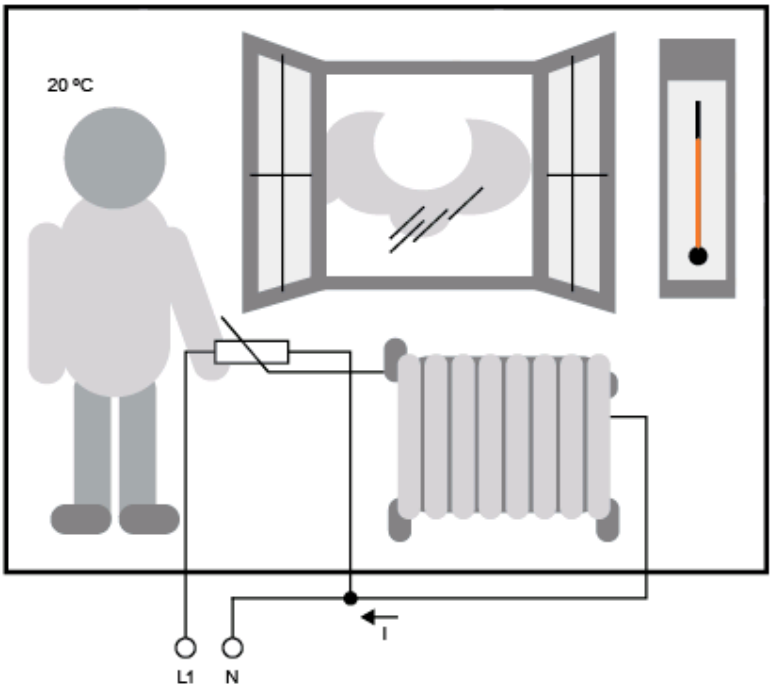
Control and regulate

In engineering, quantities can be both controlled and regulated.

When controlling, a quantity is manipulated without being able to compensate for outside influences. When regulating, a quantity is maintained at a specific value in order to compensate for outside influences.

In the following example, controlling means that the person can set the heat output at a fixed value. The heater cannot compensate for the drop in room temperature when a window is opened.

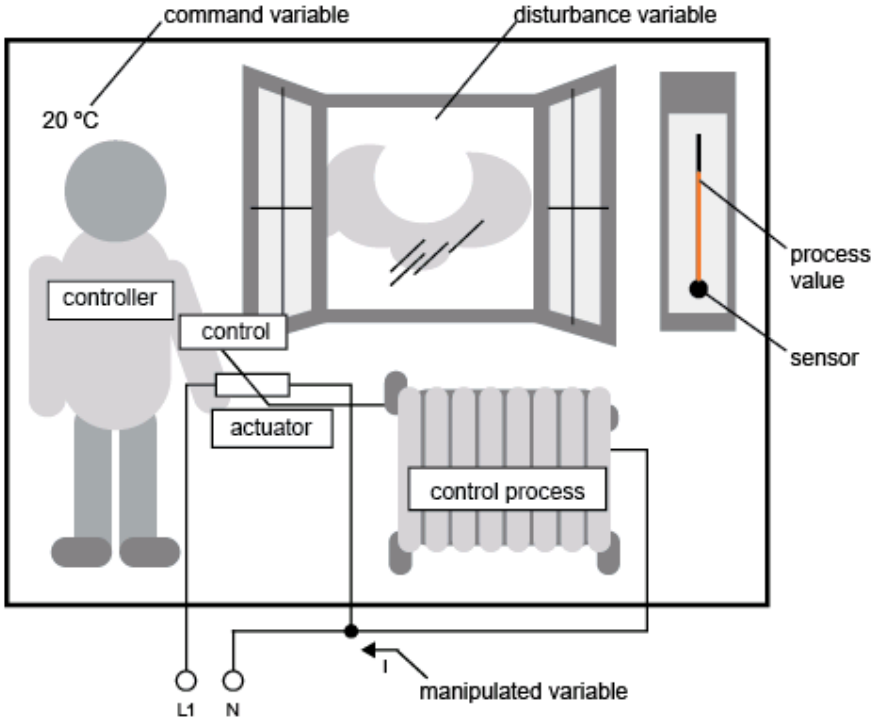
In the example below, regulating means that the person can increase the heat output if the room temperature drops to below 20 °C. If the room temperature rises above 20 °C, the heat output is reduced.



Basic concepts of regulating

In the example, the current for the electric heating is the **manipulated variable**. The changeable resistance is the **actuator**. The hand that operates the actuator is the **control**. The actual room temperature is the controlled variable or the **process value**. The desired room temperature is the **command variable** or the setpoint value. The electric heating is the **control process**. The thermometer is the **sensor**. The temperature loss from opening the window is the **disturbance variable**.

So this means that the person measures the process value (room temperature) with the sensor (thermometer), compares the process value (room temperature) with the command variable (desired room temperature) and uses the actuator (changeable resistance) to manually regulate the manipulated variable (heating current), in order to compensate for the disturbance variable (temperature drop from opening the window). The person is therefore the controller.

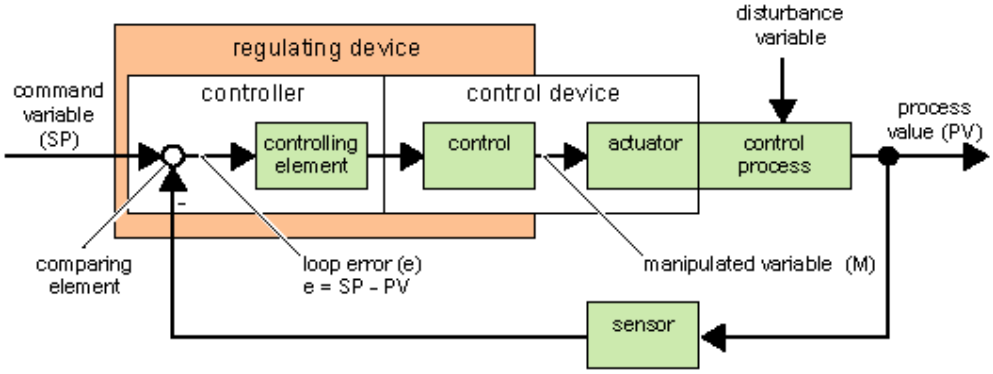


The **control device** is formed from the actuator and the control.

The control and controller together form the **regulating device**.

The following picture gives an abstract portrayal of the situation described above.

The comparing element uses the sensor to compare the command variable with the process value. If the command variables and process value deviate from one another, this results in a positive or negative loop error that in turn changes the process value.



Control loop

The process value x influences the manipulated variable M by means of the regulating device. This creates a closed circuit that is also known as a **control loop**.

If, in the example above, the window is opened, the temperature in the room drops. The person must increase the heat output of the heater. If the heat output is increased too much, it will get too hot. The person must then reduce the heat output.

If the heat output is increased or reduced too quickly, then the control loop starts to sway. The room temperature fluctuates. It is either too hot or too cold. To prevent this, the person must carefully and slowly reduce or increase the heat output.

Loop error

The loop error is the difference between the command variable and the process value. In other words: the deviation of a process value from a set value.

e = SP – PV

The loop error e brings about a change to the manipulated variable M.

The example above illustrates this very well: if, with a desired temperature of 20 °C (= command value w), the room temperature is 22 °C (= process value PV), this results in the loop error:

e = SP – PV = 20 °C - 22 °C = -2 °C

In this case, the negative sign indicates a reversing action: the heat output is reduced.

In a control loopstate of equilibrium, the loop error is zero or very small. If the command variable changes or there is a disturbance, a loop error arises. The loop error is corrected by means of the manipulated variable M.

Controller basics

Description of the individual parameters

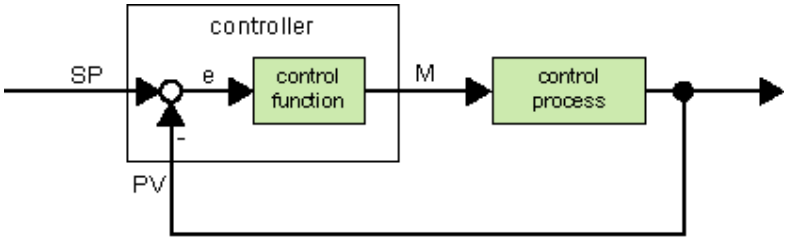
PI controller

Heating control

Ramp control

5.3.6.2 Controller basics

A controller can be simply portrayed as follows:



The comparing element and the controller function describe the conduct of the controller.

The following describes the most important types of controller. A controllerstep response tells us a lot about its conduct. The step response describes how a controller reacts to the erratic change in the process value.

There are 3 important basic types of controller:

- Proportional-action controller (P controller)
- Integral-action controller (I controller)
- Differential-action controller (D controller – wenot touching on this here)

These are combined for a real controller. For instance, the PI controller:

P Controller

A proportional-action controller (P controller) changes the manipulated variable M proportional to the loop error. The P controller works immediately. By itself it cannot drive the loop error to zero.

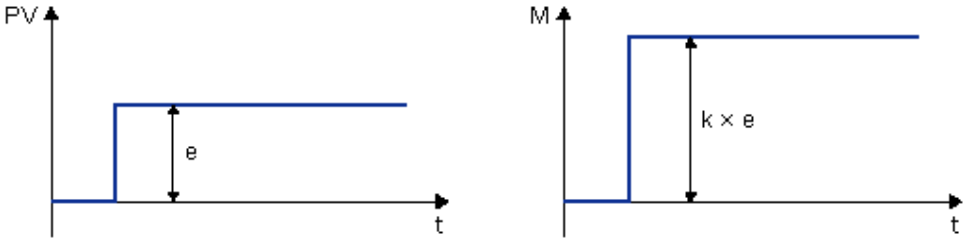
$M_{Pn} = k_P \times e_n$

M_{Pn}: Manipulated variable of the P controller at the time n

k_P: Gain of the P controller

e_n: Loop error at the time n

The following picture shows a jump in process value and step response of the controller:



Summary

The P controller has the following characteristics:

- It cannot correct faults with the control process > lasting loop error.
- It reacts immediately to a change in the process value.
- It is stable.

I Controller

An integral-action controller (I controller) changes the manipulated variable M proportional to the loop error and to the time. The I controller works by delayed action. It completely remedies a loop error.

In order to calculate the value of the manipulated variable at a period of time **n**, the time up until this period of time must be divided into small time slices. The loop errors at the end of each time slice must be added up (integrated) and they are then entered in the calculation.

$$M_{In} = k_I \times (T_S / T_I) \times (e_n + e_{n-1} + e_{n-2} + e_{n-3} + \ldots + e_0) = k_I \times (T_S / T_I) \times e_n + M_{In-1}$$

M_{In}: Manipulated variable of the I controller at the time n

M_{In-1}: Manipulated variable of the I controller at the time n-1; also called integral sum

k_I: Gain of the I controller

T_S: Sampling time, duration of a time slice

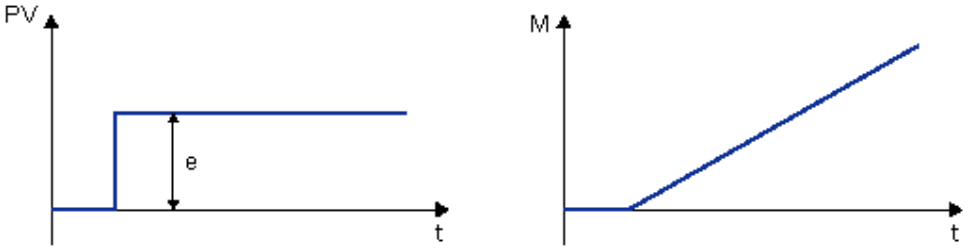
T_I: Integral time: by means of this time, the influence of the integral part is controlled on the manipulated variable, also known as integral-action time

e_n: Loop error at the time n

e_{n-1}: Loop error at the time n-1; etc.

e_0: Loop error at the beginning of the calculations

The following picture shows a jump in process value and step response of the controller:



Summary

The I controller has the following characteristics:

- It sets the process value exactly to the command variable.
- By so doing, it tends to oscillate and is unstable.
- It requires more time to carry out the control action than the P controller .

PI controller

A PI controller reduces the loop error immediately and will eventually drive the loop error to zero.

$$M_n = M_{Pn} + M_{In} = k_P \times e_n + k_I \times (T_S / T_I) \times e_n + M_{In-1}$$

M_n: Manipulated variable at the time n

M_{Pn}: Proportional part of the manipulated variable

M_{In}: Integral part of the manipulated variable

M_{In-1}: Manipulated variable of the I controller at the time n-1; also called integral sum

k_P: Gain of the P controller

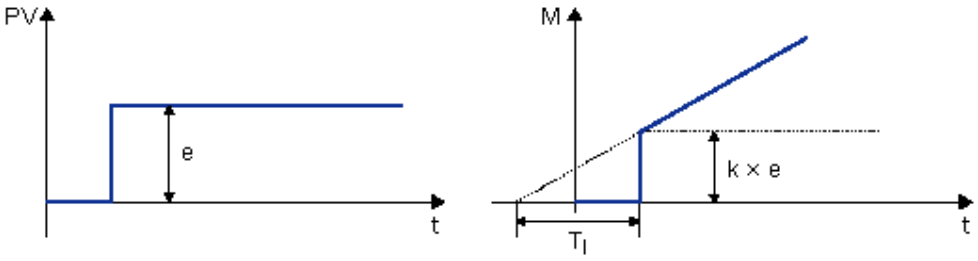
k_I: Gain of the I controller

T_S: Sampling time, duration of a time slice

T_I: Integral time; by means of this time the influence of the integral part is controlled on the manipulated variable, also known as the integral-action time

e_n: Loop error at the time n

The following picture shows a jump in process value and step response of the controller:



Summary

The PI controller has the following characteristics:

- The P controller components quickly intercept an occurring loop error.
- The I controller components can then remedy the remaining loop error.
- The controller components supplement each other so that the PI controller works quickly and precisely.

5.3.6.3 Description of the individual parameters

Controller parameters	Portrayed in LOGO!	Possible value range in the LOGO!
Mn Manipulated variable at the time n	Output of the PI controller block	0 to 1,000
kP Gain of the P part kI Gain of the I part	In the LOGO!, the parameter KC applies as an increase for the I part and the P part of the controller equally. Should you enter KC=0, then the P part of the controller switches off. In this special case, k is automatically set to 1 for the I part. If KC = 0: kP = 0 and kI = 1 If KC <> 0: kP = kI = KC	0.00 to 99.99
T_S Sampling time, duration of a time slice	Fixed	500 ms
T_I Integral time	Parameter TI, if you set this parameter to 99:59 min, then you switch off the I part of the controller.	00:01 min to 99.59 min
e_n Loop error at the time n; generally applies: e = SP – PV	Refer to SP and PV	
SP	The parameter SP is the set-value assignment w. For this parameter you can use the analog output of a different special function.	-10,000 to +20,000
PV	PV is the process value x and is calculated as follows: PV = (analog value on input * gain) + offset. You can connect the input for example by means of an analog input with a PT100 sensor.	
	The gain parameter has an effect on PV	0.0 to 10.0
	The offset parameter has an effect on PV	-10,000 to +20,000
	PV is restricted by the parameters Min. and Max.	In each case: -10,000 to +20,000

Controller parameters	Portrayed in LOGO!	Possible value range in the LOGO!
	The Dir parameter gives the action direction of the controller. Positive means: If set value > process value then the process value is increased; if set value < process value then the process value is reduced. Negative means: If set value > process value then the process value is reduced; if set value < process value then the process value is increased. e.g. heat regulation: if the set value is greater than the process value (room is too cold), the manipulated variable increases the process value.	- or +

Refer to the PI controllers - description of special function for more details (for example, switching from manual to automatic mode, parameter sets, etc.).

5.3.6.4 PI controller

Short description

A PI controller is a proportional-action and integral-action controller. You can use both proportional action and integral action individually or combined.

Connection	Description
Input **A/M**	Set the mode of the controller: 1: automatic mode 0: manual mode
Input **R**	Use the input R to reset the output AQ. As long as this input is set, the input A/M is disabled. The output AQ is set to 0.
Input **PV**	Analog value: process value, influences the output
Parameter	Sensor: Type of sensor being used **Min.:** Minimum value for PV value range: -10,000 to +20,000 **Max.:** Maximum value for PV value range: -10,000 to +20,000 **A:** Gain Value range: +- 10.00 **B:** Offset Value range: +- 10,000 **SP:** Set-value assignment value range: -10,000 to +20,000 **Mq:** Value from AQ with manual mode. Value range: 0 to 1,000 **Parameter sets:** application-related presets for KC, TI and Dir (see below) **KC:** Gain value range: 00.00 to 99.99 **TI:** Integral time value range 00:01 min to 99:59 min **Dir:** Action direction of the controller value range: + or - **p:** Number of decimal places value range: 0, 1, 2, 3
Output **AQ**	Analog output (manipulated variable) Value range for AQ: 0 to 1,000

Parameters SP and Mq

The set-value SP and the value for Mq can be provided by the actual value of another already-programmed function:

- Analog comparator: Ax – Ay
- Analog trigger: Ax
- Analog amplifier: Ax
- Analog multiplexer: AQ
- Analog ramp: AQ
- Analog math: AQ
- PI controller: AQ
- Up/Down counter: Cnt

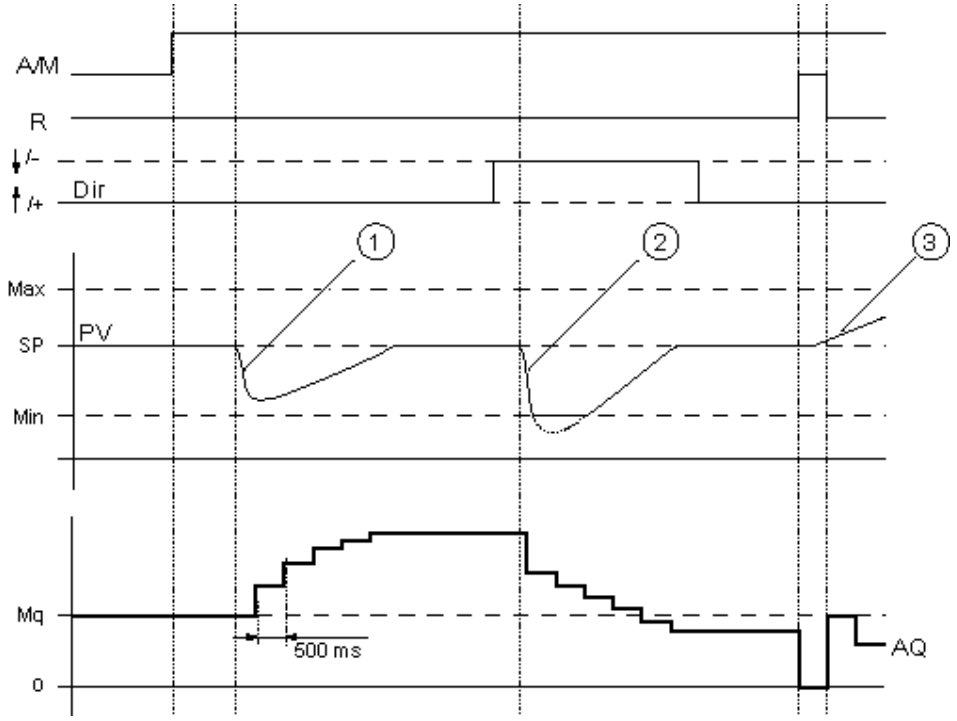
You select the required function by the block number.

Parameter P (number of decimal places)

Parameter p only applies for displaying the values from PV, SP, Min. and Max. in a message text.

Timing diagram

The nature, manner and speed with which the AQ changes depends on the parameters KC and TI. Thus, the course of AQ in the diagram is merely an example. A control action is continuous; therefore the diagram portrays just an extract.



① A disturbance causes the PV to drop, as Dir is positioned upwards, AQ increases until PV corresponds again to SP.

② A disturbance causes the PV to drop, as Dir is positioned upwards, AQ decreases until PV corresponds again to SP.

Dir is coordinated to the basic conduct of a control loop. The direction (dir) cannot be changed during the term of the function. The change in Dir here is shown for the purposes of clarification.

③ As AQ is set to 0 by means of the input R, PV changes. This is based on the fact that PV increases, which on account of Dir = upwards causes AQ to drop.

When you view the circuit program in an online test or in a simulation, LOGO!Soft Comfort displays a trend view of the analog output value of the PI controller. The trend view shows the change in the analog output value over time.

Description of Function

If the input A/M is set to 0, then the special function issues output AQ with the value that you set with parameter Mq.

If the input A/M is set to 1, then automatic mode commences. As an integral sum the value Mq is adopted, the controller function begins the calculations in accordance with the formulas given in Control and regulate basics. The updated value PV is used in the formulas.

Updated value PV = (PV * gain) + offset

If the updated value PV = SP, then the special function does not change the value of AQ.

Dir = upwards/+ (timing diagram numbers 1 and 3)

- If the updated value PV > SP, then the special function reduces the value of AQ.
- If the updated value PV < SP, then the special function increases the value of AQ.

Dir = downwards/- (timing diagram number 2)

- If the updated value PV > SP, then the special function increases the value of AQ.
- If the updated value PV < SP, then the special function reduces the value of AQ.

With a disturbance, AQ increases or decreases until the updated value PV again corresponds to SP. The speed with which AQ changes depends on the parameters KC and TI.

If the input PV exceeds the parameter Max., then the updated value PV is set to the value of Max. If the PV falls short of the parameter Min., then the updated value PV is set to the value of Min.

If the input R is set to 1, then the AQ output is reset. As long as R is set, the input A/M is disabled.

Sampling time

The sampling time is fixed at 500 ms.

Parameter sets

In order to simplify the use of the PI controller, the parameters for KC, TI and Dir are already given as sets for the following applications:

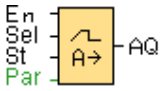
Parameter set	Application example	Parameter KC	Parameter TI (s)	Parameter Dir
Temperature fast	Temperature, cooling control of small spaces; small volumes	0.5	30	+
Temperature slow	Heating, ventilation, temperature, cooling control of large spaces; large volumes	1.0	120	+
Pressure 1	Quick pressure change, compressor control	3.0	5	+
Pressure 2	Slow pressure change, differential pressure control (flow controller)	1.2	12	+
Full level 1	Vat and/or reservoir filling without drain	1.0	99:59	+
Full level 2	Vat and/or reservoir filling with drain	0.7	20	+

Characteristics when configuring

Observe the Control and regulate basics.

Overview

5.3.6.5 Ramp control



Short Description

The Analog Ramp instruction allows the output to be changed from the current level to the selected level at a specified rate.

Connection	Description
Input **En**	A change in the status from 0 to 1 at input En (Enable) applies the start/stop level (Offset "B" + StSp) to the output for 100 ms and starts the ramp operation to the selected level. A change in the status from 1 to 0 immediately sets the current level to Offset "B", which makes output AQ equal to 0.
Input **Sel**	Sel = 0: The step 1 (level 1) is selected. Sel = 1: The step 2 (level 2) is selected. A change in status of Sel causes the current level to start changing to the selected level at the specified rate.
Input **St**	A change in the status from 0 to 1 at input St (Decelerated Stop) causes the current level to decrease at a constant rate until the start/stop level (Offset "B" + StSp) is reached. The start/stop level is maintained for 100 ms and then the current level is set to Offset "B", which makes output AQ equal to 0.
Parameter	**Level1 and Level2:** Levels to be reached; value range for each level: -10,000 to +20,000 **MaxL:** Maximum value that must not be exceeded. Value range: -10,000 to +20,000 **StSp:** Start/Stop offset: value that is added to Offset "B" to create the start/stop level. If the Start/Stop offset is 0, then the start/stop level is Offset "B"). Value range: 0 to +20,000 **Rate:** Speed with which level 1, level 2 or 0ffset is reached. Steps/seconds are issued. Value range: 1 to 10,000 **A:** Gain Value range: 0 to 10,00 **B:** Offset Value range: +- 10.000 **p:** Number of decimal places Value range: 0, 1, 2, 3
Output **AQ**	The output AQ is scaled using the formula: (Current Level - Offset "B") / Gain "A" Note: When AQ is displayed in parameter mode or message mode, it is displayed as an unscaled value (engineering units: current level). Value range for AQ: 0...+32767

Parameters Level1 and Level2

The level parameters Level1 and Level2 can be provided by the value of another already-programmed function:

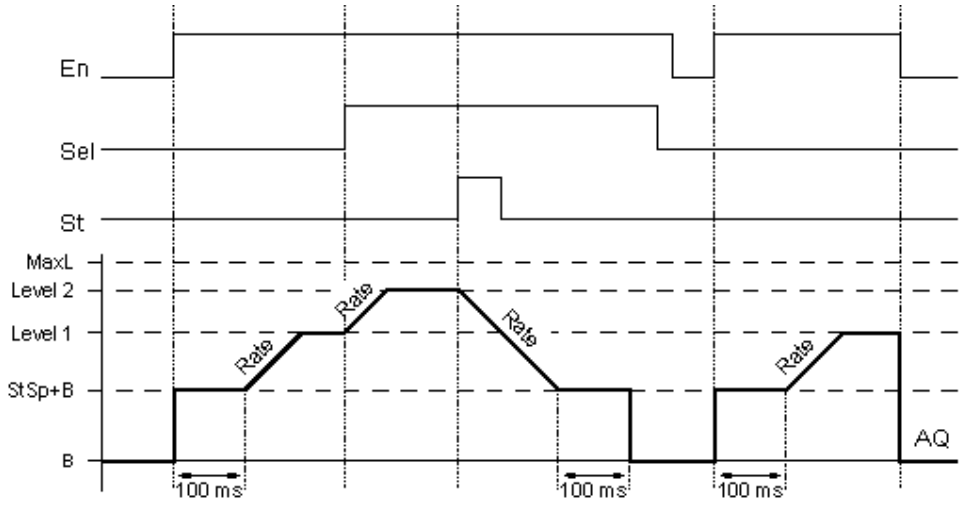
- Analog comparator: Ax – Ay
- Analog trigger: Ax
- Analog amplifier: Ax
- Analog multiplexer: AQ
- Analog ramp: AQ
- Analog math: AQ
- PI controller: AQ
- Up/Down counter: Cnt

You select the required function by the block number.

Parameter p (number of decimal places)

Parameter p only applies for displaying the values of AQ, level 1, level 2, MaxL, StSp, and Rate in a message text.

Timing diagram for AQ



Description of function

If the input En is set, then the function sets the value StSp + Offset "B" for 100 ms.

Then, depending on the connection of Sel, the function runs from the level StSp + Offset "B" to either level 1 or level 2 at the acceleration set in Rate.

If the input St is set, the function runs to a level of StSp + B at the acceleration set in Rate. Then the function holds the level at StSp + Offset "B" for 100 ms. After 100 ms, the level is set to Offset "B". output AQ. The scaled value (output AQ) is 0.

If the input St is set, the function can only be restarted once the inputs St and En have been reset.

If input Sel has been changed, depending on the connection of Sel, the function runs from the current target level to the new target level at the rate that is specified.

If the input En is reset, the function immediately sets the current level to Offset "B".

The current level is updated every 100 ms. Note the relationship between output AQ and the current level:

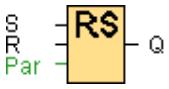
Output AQ = (current level – Offset "B" / Gain "A")

Particular characteristics to be noted when configuring

For help on analog block parameters, refer to the Analog value processing section.

5.3.7 Miscellaneous

5.3.7.1 Latching relay

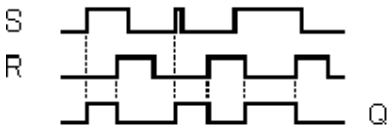


Short description

A signal at input S sets output Q. A signal at input R resets output Q.

Connection	Description
Input **S**	Set output Q with a signal at input S (Set).
Input **R**	Reset output Q with a signal at input R (Reset). Output Q is reset if S and R are both set (reset has priority over set).
Parameter	**Retentivity** set (on) = the status is retentive in memory.
Output **Q**	Q is set with a signal at input S and remains set until it is reset with signal at input R.

Timing diagram



Description of the function

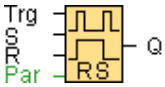
The latching relay represents a simple binary memory logic. The output value depends on the input states and the previous status at the output.

Logic table of the latching relay:

S	R	Q	Remark
0	0	x	Status unchanged
0	1	0	Reset
1	0	1	Set
1	1	0	Reset

When retentivity is enabled, the output signal corresponds with the signal status prior to the power failure.

5.3.7.2 Pulse relay



Short description

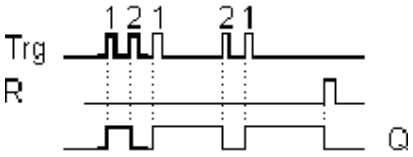
The output is set and reset with a short one-shot at the input.

Connection	Description
Input **Trg**	You switch output Q on or off with a signal at input Trg (Trigger) input.
Input **S**	A one-shot at input S (Set) sets the output to logical 1.
Input **R**	A one-shot at input R (Reset) resets the output to logical 0
Parameter	**Selection:** RS (input R priority), or SR (input S priority) **Retentivity** set (on) = the status is retentive in memory.
Output **Q**	Q is switched on with a signal at Trg and is reset again at the next Trg pulse, if both S and R = 0.

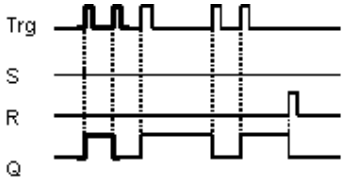
0BA0-0BA3:

The special function does not have an S input and priority preselection.

The following applies to output Q:
Q is switched on with a signal at Trg and is switched off again with the next signal at Trg or R.



Timing diagram



Description of the function

The status of output Q changes with each 0 to 1 transition at input Trg and if both S and R = 0, that is, the output is switched on or off.

Input Trg does not influence the SFB when S = 1 or R = 1.

A one-shot at input S sets the pulse relay, that is, the output is set to logical 1.

A one-shot at input R resets the pulse relay to its initial state, that is, the output is set to logical 0.

Either the input R takes priority over input S (the signal at input S has no effect as long as R = 1), or the input S takes priority over input R (the signal at input R has no effect as long as S = 1), depending on your configuration.

0BA0-0BA3:

The valid function is:

The status at output Q is toggled with each 0 to 1 transition at input Trg, that is, the output is switched on or off.

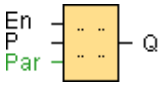
You reset the pulse relay to its initial state with a one-shot at input, that is, the output is reset to 0.

The pulse relay is reset and output Q = 0 after power on or by a reset signal.

Caution

If Trg = 0 and Par = RS, the "Pulse relay" SFB corresponds with the "Latching relay" SFB function.

5.3.7.3 Message text (LOGO! 0BA6)



Short description

This function displays message texts and parameters of other blocks on the LOGO! Display or LOGO! TD when LOGO! is in RUN mode.

The LOGO! 0BA6 device series supports many new message text features that LOGO! 0BA5 devices and earlier did not support. You can choose, however, whether to use LOGO! 0BA6 message text functions blocks with the new features or LOGO! 0BA5 message text function blocks in your circuit program. This selection is on the File -> Message Text Settings dialog, along with other global settings. You cannot mix and match LOGO! 0BA5 message text function blocks and LOGO! 0BA6 message text function blocks in your circuit program.

Connection	Description
Input **En**	A 0 to 1 transition at En (Enable) triggers the output of the message text.
Input **P**	P is the priority of the message text. 0 is the lowest, 127 the highest priority. Ack: Acknowledgement of the message text
Parameter	**Text:** Input of the message text **Par:** Parameter or actual value of another, already configured , which can be displayed numerically or as a bar graph (see "Visible parameters or actual values") **Time:** Shows the continuously updated time-of-day **Date:** Shows the continuously updated date **EnTime:** Display of the time of the 0 to 1 transition of the signal at input En **EnDate:** Display of the date of the 0 to 1 transition of the signal at input En **I/O Status Names:** Display of a digital input or output status name, for example "On" or "Off". **Analog Input:** Display of analog input value to the shown in message text and updated according to the analog time.
Output **Q**	Q remains set as long as the message text is queued.

Settings

In addition to the function block inputs and the parameters of the message text, the following settings provide additional control over the display of message texts:

- **Character set selection:** You can choose to compose a message text from characters from the primary character set or the secondary character set. You set the two character sets from either the "Msg Config" menu on the LOGO! Basic module, or by using the LOGO!Soft Comfort File->Message Text Settings menu command.
- **Message destination:** You can choose to display the message text on the LOGO! Display, the LOGO! TD, or both.
- **Tick settings:** The message text can tick or not tick, meaning it can scroll on and off the display. The tick capabilites and your choices are described in more detail below.

Description of the function

With a 0 to 1 transition of the signal at input En, and with LOGO! in run mode, the LOGO! Display and/or LOGO! TD displays your configured message text.

When the "Acknowledgement message" checkbox is not selected, the message text is hidden with a 1 to 0 signal transition at input En.

When the "Acknowledgement message" checkbox is selected, then after input En is reset to 0, the message text is displayed until acknowledged by pressing the OK button. The message text cannot be acknowledged as long as input En is high.

If several message text functions were triggered with En=1, the message with the highest priority (0 = lowest, 127 = highest) is displayed. This also implies that a new message text is only displayed if its priority is higher than that of previously enabled message texts.

If the circuit program uses flag M27, then whenever M27=0 (low) LOGO! only displays message texts that are in the primary character set (Character Set 1). If M27=1 (high), then LOGO! only displays message texts that are in the secondary character set (Character Set 2).

After a message text is disabled or acknowledged, the display automatically shows the previously active message text with the highest priority.

You can change between the display in RUN mode and the message texts by means of the and buttons.

Particular characteristics to be noted when configuring

1 **Block name area**

Here you can provide a name for the block

2 **Settings area**

Here you can configure the following settings:·

- Priority of the message text
- Acknowledge Message check box: if set requires a message to be acknowledged in order to be closed
- Character set selection for the message text

3 **Tick area**

Here you define the tick parameters for the message text:

- Character by Character tick format
- Line by Line tick format
- Tick enabled checkbox for each display line

4 **Message destination area**

Here you choose whether the target destination for the message is the LOGO! Display, the LOGO! TD, or both.

5 **Message area**

Here you arrange the message text. The information that you enter in this area

corresponds to what will be displayed on the LOGO! Display or LOGO! TD.

Above this area are additional buttons:

 **Delete button:** for deleting entries from the message area

 **Special characters button:** for inserting special characters into the message area

 **Bar Graph button:** for placing a horizontal or vertical bar graph into the message area

AI button: for placing an analog input value into the message area

ON/OFF button: for specifying a digital value to be represented by one of two strings corresponding to the 0 state and 1 state, for example "OFF" and "ON".

Edit manually button: for using the static editor to add, move, or delete message text elements without changing the position of any other elements

⑥ **Block area**

Here is where you select blocks from all of the blocks in the circuit program. You can subsequently choose parameters from these blocks to display in the message text.

⑦ **Block parameters area**

Here you can select the parameters to be displayed in the message text from the block that is selected in the Blocks area.

⑧ **Insert Parameter button**

This button is for inserting a selected block parameter into the message text.

To arrange the message text

The message area shows a grid of four lines and character positions. The message text configuration area is 24 characters wide for Western European character sets or 16 characters wide for Asian character sets. Either way, the character width of each line is double that of the LOGO! Display or LOGO! TD. Message lines that are longer than the width of the actual display can be set to "tick". In the message area, LOGO!Soft Comfort indicates the region corresponding to the visible area of the LOGO! Display or LOGO! TD in one color, and the region that can only be shown by using the message ticking feature in another color.

To configure the content of a message text, follow these steps:

1. From the "Blocks" area, select the block whose parameters you want to output.
2. Drag and drop the parameters required from the "Block parameters" to the "Message Text" area. You can also use the "Insert " button to insert a parameter value.
3. In the "Message Text" area, you can add parameter data, time or date values from the block parameter area as required and enter text. To enter text, select the character set for the message text, and then type. You can also use the buttons above the message text area to add special characters, bar graphs, analog input values, and names for digital I/O states.

Message text character set

LOGO! 0BA6 supports five character sets for messages. Of these, you can select two for the display of message texts with the File -> Message Text Settings menu command or from the "Msg Config" menu of LOGO!. Of the fifty possible message texts that you can configure, you can select any number of them to be from the first language and any number from the second language. For example, you could configure fifty message text function blocks that have a single message text for Character Set 1. Alternatively, you could configure twenty-five message text function blocks, each of which has two message texts: one for Character Set 1 and one for Character Set 2. Any combination is valid such that the total does not exceed fifty.

The Message text dialog displays the character sets that are currently available for your message text, as configured in the message text settings. To use a character set, select the "Enabled" check box, and the button for the character set. Characters that you subsequently type will be from the character set that you enabled and selected. If you deselect the "Enabled" check box for a character set, LOGO!Soft Comfort will prompt you for confirmation and then delete the message text, if it exists, that corresponds to that character set.

The character set of a message text is independent of the language setting for the LOGO! Display menus. They can be different.

Chinese character set

The LOGO! Basic module and LOGO! TD support the Chinese character set (GB-2312) for the People's Republic of China. The devices use Microsoft Windows encoding for this character set. The Windows encoding allows the devices to display the same characters as shown in the LOGO!Soft Comfort message text editor when you are using a Chinese emulator or a Chinese version of Microsoft Windows.

The Chinese character set requires a Chinese version of Windows or a Chinese emulator to properly display Chinese characters in the LOGO!Soft Comfort message text editor. You must start the Chinese emulator before you open the the message text function block in LOGO!Soft Comfort.

Message ticker

You can configure a message text to tick, or to not tick. Two types of message ticking exist:

- Character by character
- Line by line

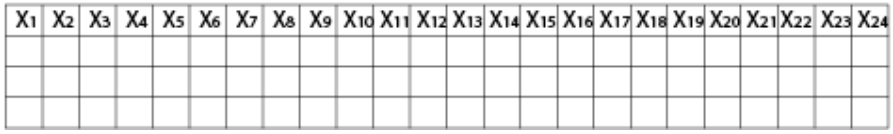
Messages that tick character by character scroll off the characters of the message line one character at a time to the left with the additional characters scrolling in one at a time from the right. The time interval for the tick is specified by the TickTime message text setting.

Messages that tick line by line by line scroll one half of the message off the display to the left with the second half of the message scrolling in from the right. The time interval for the tick is ten times the TickTime parameter. The two halves of the message simply alternate on the LOGO! Display or LOGO! TD.

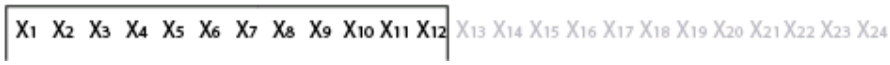
The tick time is the interval of time by which a character or a line of text ticks off the screen. The tick time is a global message text parameter for all message texts.

Example: Tick Message Character by Character

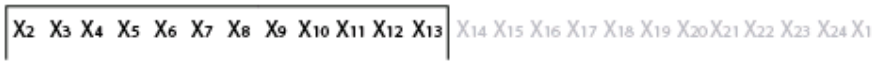
The following illustration shows the configuration in LOGO!Soft Comfort of a one-line, 24-character message text:



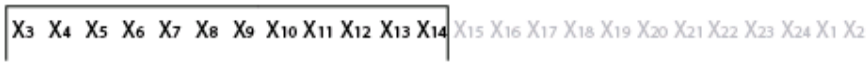
If you set this message to tick "character by character" with a tick interval of 0.1 seconds, then the initial appearance of this message line on the LOGO! Display or LOGO! TD is as shown in this illustration:



After 0.1 second, one character of the message line ticks. The message appears as follows on the LOGO! Display or LOGO! TD:

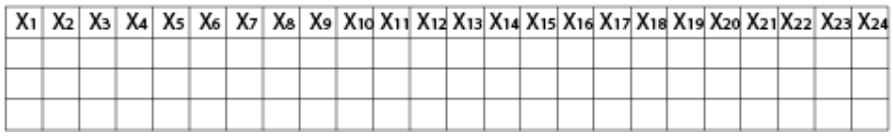


After the next 0.1 second, another character of the message line ticks. The message appears as follows on the LOGO! Display or LOGO! TD:

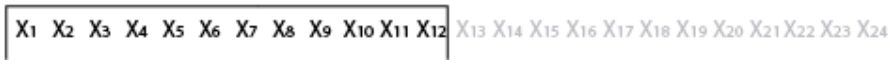


Example: Tick Message Line by Line

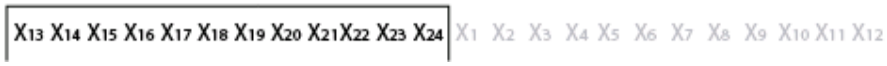
The following example uses the same message configuration as the previous example:



If you set this message to tick "line by line" with a tick interval of 0.1 seconds, then the initial appearance of this message on the LOGO! Display or LOGO! TD is the left half of the message as shown in this illustration:



After 1 second (10 x 0.1 second), the message ticks to show the right half of the message as shown in this illustration:



The screen display alternates between the two message halves every second.

You can configure each individual line of a message text to tick or not to tick at all. The "character by character" or "line by line" setting applies to all lines that you configure to tick. Select the check box beside a line number to enable ticking for that line.

Bar graphs

You can specify a bar graph representation for the actual value of any other function block in your circuit program. The bar graph can be horizontal or vertical on the LOGO! Display or LOGO! TD. You can configure up to four bar graphs per message text.

Use the bar graph button  in the message text area to place a bar graph in the message text area. From the Bar Setting dialog you must provide the following information:

- Block from your circuit program that the bar graph represents
- Minimum and maximum value for the bar graph: LOGO! will calculate the length or height of the bar graph by scaling the actual value between the minimum and maximum values.
- Orientation of the bar graph: horizontal or vertical
- Width or height of the bar graph, in character spaces

Example:

Consider a bar graph in a message text with the following characteristics:

- Configured bar graph length: 4 character spaces
- Orientation: Horizontal
- Configured minimum value: 1000
- Configured maximum value: 2000
- Actual value: 1750

The resulting bar graph will be 3 character spaces long.

Text representation of digital I/O states

You can assign names such as "On" or "Off" to the two states of a digital input or output. With the LOGO! 0BA6 series, you can display this name of a digital I/O state in a message text. The maximum number of characters for a state name is eight for Western language character sets; for Asian language character sets the number of characters is four. Use the ON/OFF button in the message text area to define names for the two states of a digital input or output to be used in the message text.

Within a single message text, you can configure up to four digital I/O state name displays.

Within one circuit program, a maximum of 20 I/O state names can be used in message text function blocks.

Display of remaining timer time

With the LOGO! 0BA6 series, you can display the remaining time of a timer in a message text. Prior to this feature you could display the current elapsed time of a timer and the timer parameters.

When placed in a message text, the remaining time will show the time that remains before the timer expires. For timers with multiple timer values (for example on-delay time, off-delay time), you can display the remaining time of each one in a message text.

Display of analog inputs

You can also select analog inputs to be displayed in a message text. Use the AI button in the message text area to select a specific AI to place in the message text area.

If you have analog inputs in a message text, the global message text setting for the Analog input filter timer specified how often the message text is updated with current values. The refresh time choices are 100 ms, 200 ms, 400 ms, 800 ms, and 1000 ms. If you have more than one analog input in a message text, the refresh rate applies to them all.

Static editor (Edit manually)

LOGO!Soft Comfort provides a static editor for message texts that can help when you need to reposition text elements. For example, it has a recycle bin area where you can temporarily move message text elements in order to rearrange the position of elements on the display area. You can move elements up, down, left, or right without changing the position of any other elements.

To use the static editor, click the "Edit manually" button above the message text area. You will also be prompted to edit manually if you try to place or move elements in the message area that have a position conflict with existing elements.

Simulation mode

Layout of message texts

Overview

0BA0-0BA3:

Maximum number of message texts: 5

Not supported: message ticker, bar graphs, analog inputs, I/O status names, and remaining timer time.

0BA4-0BA5:

Maximum number of message texts: 10
Not supported: message ticker, bar graphs, analog inputs, I/O status names, and remaining timer time.

Restrictions

The following restrictions apply to message text function blocks:

- Up to 50 message text functions are available.
- Up to 32 total bar graphs in message texts are available.
- LOGO!Soft Comfort supports all of the defined features for message texts. When programming directly on the LOGO! device, you can only program a limited number of the message text features. See the LOGO! manual for a description of message text programming from the LOGO! device.
- Each message line can contain 24 characters (Western language character sets) or 16 characters (Asian language character sets). Within a message text the following limitations apply:

  - Maximum number of parameters: 4
  - Maximum number of bar graphs: 4

- Maximum number of I/O status names:4
- Maximum number of time/date values: 4
- Maximum number of analog inputs: 2

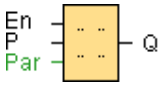
Particular characteristics to be noted when configuring

The message text can be configured in the block properties dialog. You can enter up to four lines for each message text (the text display of the LOGO! Display and LOGO! TD has four rows) and set the priority. You can move to the next line using the cursor keys or the mouse. Hit the [ENTER] key to confirm all your entries in the block properties dialog and to close the dialog.

You can also specify the actual values of other blocks in the text lines. To do so, select the relevant block from the **Block** dialog. A **Parameter** dialog opens to display a list of all parameters available for the selected block. The block parameter you select in this dialog is written to the selected text line. The actual parameter value is now included when you call the message text.

Set the "Acknowledge message" attribute to specify whether a message is be acknowledged before it is closed.

5.3.7.4 Message text (LOGO! 0BA5)



The description of the device series 0BA3 and earlier is found below.

Short description

This function displays message texts and parameters of other blocks on the LOGO! Display when LOGO! is in RUN mode.

Note: The LOGO! 0BA6 device series supports many new message text features that LOGO! 0BA5 devices and earlier did not support. You can choose, however, whether to use LOGO! 0BA6 message text functions blocks with the new features or LOGO! 0BA5 message text function blocks in your circuit program. This selection is on the File -> Message Text Settings dialog, along with other global settings. You can also use the button at the bottom of the LOGO! 0BA5 message dialog to change your message text functions to the LOGO! 0BA6 style with the new features. You cannot mix and match LOGO! 0BA5 message text function blocks and LOGO! 0BA6 message text function blocks in your circuit program.

Connection	**Description**
Input **En**	A 0 to 1 transition at En (Enable) triggers the output of the message text.
Input **P**	P is the priority of the message text. 0 is the lowest, 30 the highest priority. Ack: Acknowledgement of the message text
Parameter	**Text:** Input of the message text **Par:** Parameter or actual value of another, already configured function (see "Visible parameters or actual values") **Time:** Shows the continuously updated time-of-day **Date:** Shows the continuously updated date **EnTime:** Shows the time of the 0 to 1 transition **EnDate:** Shows the 0 to 1 transition of the date
Output **Q**	Q remains set as long as the message text is queued.

Description of the function

With a 0 to 1 transition of the signal at input En, the display outputs your configured message text (actual value, text, TOD, date) in RUN mode.

Acknowledgement disabled (Ack = Off):

The message text is hidden with a 0 to 1 signal transition at input En.

Acknowledgement enabled (Ack = On):

After input En is reset to 0, the message text is displayed until acknowledged by pressing the OK button. The message text cannot be acknowledged as long as input En is high.

If several message text functions were triggered with En=1, the message with the highest priority (0 = lowest, 30 = highest) is displayed. This also implies that a new message text is only displayed if its priority is higher than that of previously enabled message texts.

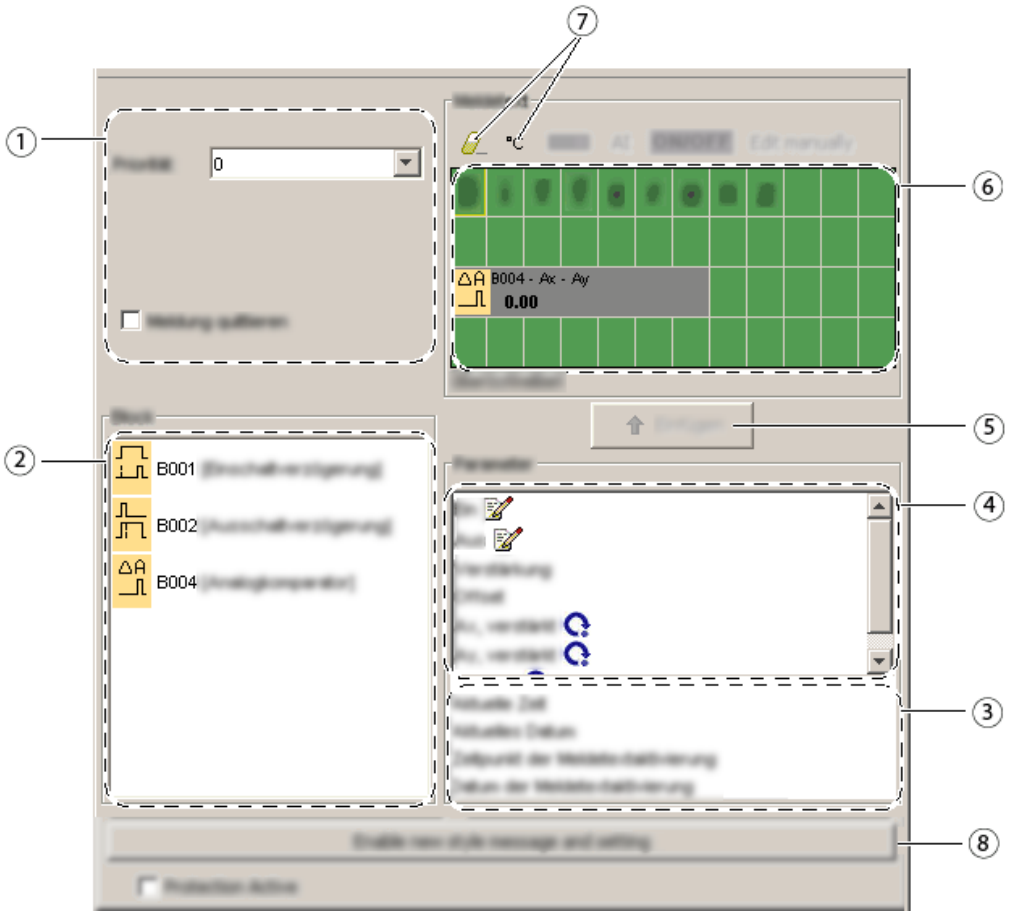
After a message text is disabled or acknowledged, the function automatically shows the previously active message text that takes the highest priority.

You can change between the display in RUN mode and the message texts by means of the and buttons.

Restrictions

Up to 10 message text functions are available.

Particular characteristics to be noted when configuring



① **"General" area**

Here you will find the following settings:·

- Priority of the message text
- Check box for message text acknowledgement

② **"Blocks" area**

Shows a list of all the circuit program blocks and their parameters.

③ **"General parameters" area**

Shows general parameters such as the current date.

④ **"Block parameters" area**

Shows the parameters of a block selected from the "Blocks" area which you can output in the message text.

⑤ **"Insert" button**

Button for inserting a parameter selected from the **"Block parameters"** or **"General parameters"** area into the message text.

⑥ **"Messages" area**

You arrange the message text in this area. Information entered in this area corresponds with that on the LOGO! display.

⑦ **"Delete" button**

Button for deleting entries from the **"Messages"** area

"Special characters" button

Button for inserting special characters in the **"Messages"** area

⑧ **"Enable new style message and setting"**

Button to switch to LOGO! 0BA6 style message texts with new features. You must then complete the File -> Message Text Settings dialog to configure settings for LOGO! 0BA6 style message texts. All message texts will be in the LOGO! 0BA6 style after this selection.

To arrange the message text

1. From the **"Blocks"** area, select the block whose parameters you want to output.
2. Drag and drop the parameters required from the **"Block parameters"** to the **"Messages"** area. You may also use the **"Insert"** button to do so.
3. In the **"Messages"** area, you can add parameter data as required.

0BA0-0BA3:

The following specifications apply:

Short description

Display of a configured message text in RUN mode

Connection	Description
Input En	The message text is output with a 0 to 1 transition at input En (Enable).
Parameter P	P is the priority of the message text. 0 is the lowest, 9 the highest priority.
Parameter	Par: Parameter or actual value of another, already programmed function (see "Displayable parameters or actual values")
Output Q	Q remains set as long as the message text is queued.

Description of the function

In RUN mode, a 0 to 1 transition of the signal at input En triggers the output of your configured message text on the display. The message text is closed after a 1 to 0 transition at input En and if the acknowledgment attribute is not set. If the acknowledgment attribute is set, the message text is not closed until input En=0 and the message is acknowledged at the LOGO! with OK. The status at output Q remains 1 as long as the message text is displayed.

Of several message text functions triggered with En=1, the one with the highest priority is displayed. Low-priority messages can also by displayed by pressing the button on the LOGO!.

You can switch between the standard display and the message text display by means of the LOGO! buttons and .

Restrictions

Up to five message text functions are available.

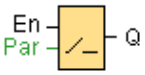
Particular characteristics to be noted when configuring

The message text can be configured in the block properties dialog. You can enter up to four lines for each message text (the text display of the LOGO! has four rows) and set the priority. You can move to the next line using the cursor keys or the mouse. Hit the [ENTER] key to confirm all your entries in the block properties dialog and to close the dialog.

You may also enter the actual values of other blocks in the text lines. To do so, select the relevant block from the **Block** dialog. A **Parameter** dialog opens to display a list of all parameters available for the selected block. The block parameter you select in this dialog is written to the selected text line. The actual parameter value is now included when you call the message text.

Set the "Acknowledge message" attribute to specify whether a message is be acknowledged before it is closed.

5.3.7.5 Softkey



Short description

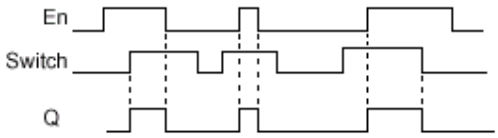
This SFB provides the action of a mechanical pushbutton or switch.

Connection	Description
Input **En**	Output Q is set with a 0 to 1 signal transition at input En (Enable) and if in addition, 'Status=On' has been confirmed in configuration mode.
Parameter	**Switch:** Sets either a momentary pushbutton action for one cycle (On) or a switching action of the function (Off). **Status**: On or Off state that is applied in the initial cycle after program startup, if retentivity is not set. **Retentivity** set (on) = the status is retentive in memory.
Output **Q**	Output Q remains set 1, as long as En=1 and the status at the parameter Type = Switch and Status = On. Output Q is set for the duration of one cycle if EN=1. Switch = On (momentary pushbutton) and Status = On.

Factory state

The default of the Switch parameter is switching action.

Timing diagram



Description of the function

The output is set when input En is set and the 'Status' parameter is set to 'On' and confirmed with OK. This action is performed irrespective of a configured switch or pushbutton function.

The output is reset to '0' in the following three cases:

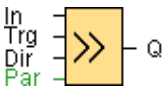
- With a 1 to 0 signal transition at input En
- When a pushbutton function is configured and one cycle has expired after its actuation
- When the 'Status' parameter sets the 'Off' status in configuration mode, and this has been confirmed with OK

Particular characteristics to be noted when configuring

The softkey can be used both with momentary pushbutton or switching action. At the status parameter you can define the on (actuated) or off state for the switch/pushbutton.

If the softkey is assigned a pushbutton action, the output is always set for the duration of one cycle with a 0 to 1 transition at input En when the pushbutton is in on state, or if the pushbutton state changes from Off to On when En=1.

5.3.7.6 Shift register

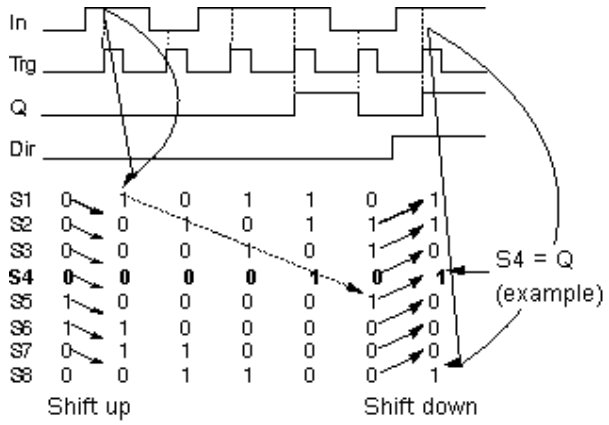


Short description

The shift register function can be used to read an input value and to shift the bits. The output value corresponds with the configured shift register bit. The shift direction can be changed at a special input.

Connection	Description
Input **In**	The function when started reads this input value.
Input **Trg**	The SFB is started with a positive edge (0 to 1 transition) at input Trg (Trigger). A 1 to 0 transition is irrelevant.
Input **Dir**	You define the shift direction of the shift register bits S1...S8 at the Dir input: Dir = 0: shift up (S1 >> S8) Dir = 1: shift down (S8 >> S1)
Parameter	**Shift register bit** that determines the value of output Q. Possible settings: S1 ... S8 **Retentivity** set (on) = the status is retentive in memory.
Output **Q**	The output value corresponds with the configured shift register bit.

Timing diagram



Description of the function

The function reads the value of input In with a positive edge (0 to 1 transition) at input Trg (Trigger).

This value is written to shift register bits S1 or S8, depending on the set shift direction:

- Shift up: S1 accepts the value of input In; the previous value of S1 is shifted to S2, S2 is shifted to S3, etc.
- Shift down: S8 accepts the value of input In; the previous value of S8 is shifted to S7, S7 is shifted to S6, etc.

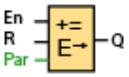
Q outputs the value of the configured shift register bits.

If retentivity is not enabled, the shift function restarts at S1 or S8 after a power failure.

Note

The special function shift register can be used only once in the circuit program.

5.3.7.7 Analog math error detection



Short description

The analog math error detection block sets an output if an error has occurred in the referenced analog math function block.

Connection	Description
Input **En**	Enable the analog math error detection function block.
Input **R**	Reset the output.
Parameter	**Referenced FB:** block number of an analog math instruction **Error to detect:** Zero division, Overflow, or Zero division OR Overflow. **Auto Reset:** Reset the output when the failure condition clears.
Output **Q**	Q is set high if the error to detect occurred in the last execution of the referenced analog math function block.

0BA1- 0BA5: The analog math error detection function block did not exist prior to 0BA6.

Parameter Referenced FB

The value for the Referenced FB parameter references the block number of an already-programmed analog math function block.

Description of the function

The analog math error detection block sets the output when the referenced analog math function block has an error. You can program the function to set the output on a zero division error, an overflow error, or when either type of error occurs.

If you select the Automatically reset checkbox, the output is reset prior to the next execution of the function block. If not, the output retains its state until the analog math error detection block is reset with the R parameter.

In any scan cycle, if the referenced analog math function block executes before the analog math error detection function block, the error is detected in the same scan cycle. if the referenced analog math function block executes after the analog math error detection function block, the error is detected in the next scan cycle.

Analog math error detection logic table

In the table below, Error to Detect represents the parameter of the analog math error detection instruction that selects which type of error to detect. Zero represents the zero division bit set by the analog math instruction at the end of its execution: 1 if the error ocurred, 0 if not. OF represents the overflow bit set by the analog math instruction: 1 if the error occurred, 0 if not. Zero division OR Overflow represents the logical OR of the zero division bit and the overflow bit of the referenced analog math instruction. Output (Q) represents the output of the analog math error detection function. An “x” indicates that the bit can be either 0 or 1 with no influence on the output.

Error to Detect	**Zero**	**OF**	**Output (Q)**
Zero division	1	x	1
Zero division	0	x	0
Overflow	x	1	1
Overflow	x	0	0
Zero division OR Overflow	1	0	1
Zero division OR Overflow	0	1	1
Zero division OR Overflow	1	1	1
Zero division OR Overflow	0	0	0

If the Referenced Analog Math FB is null, then the output is always 0.

5.3.7.8 Additional functions of the LAD Editor

In LAD circuit programs the AND with edge evaluation instruction and the NAND with edge evaluation instruction are available in the group of miscellaneous functions.

5.4 Circuit programs

5.4.1 Circuit programs - Introduction

With LOGO!Soft Comfort you can start right away to design your own circuit programs. The tutorial section provides you with detailed information about the creation and simulation of circuit programs.

You first use the LOGO!Soft Comfort software to write your programs, and secondly let LOGO!Soft Comfort calculate the minimum LOGO! version that you need to put your ideas into practice!

No particular program settings are required.

The type of LOGO! device you can use to put your ideas into practice depends on a number of factors:

- The number of I/Os to be used
- Memory requirements of the circuit program
- The use of particular SFBs

5.4.2 LOGO! Hardware

LOGO! hardware series

LOGO!Soft Comfort lets you create programs for LOGO! devices of various series. Differences are found in the performance, memory space, number of program blocks (for example, flag blocks) and in the structure of the devices.

Because there are device-specific differences with regard to the functions, you are forced not only to distinguish between the LOGO! versions, but also to take their version status into account. You can identify the version status by the suffix of the LOGO! order number.

The current LOGO! versions belong to the 7th generation. They can be identified by the last digits of their order number: 0BA6.

You can identify the first three generations of LOGO! devices by the 0, 1 and 2 suffix. The major difference in the first three generations of LOGO! lies in their modular structure. Long and AS-Interface versions are no longer available as of version 0BA3. The standard devices are now equipped with an expansion interface, to which you can connect expansion modules (digital/analog and bus modules) for adapting the LOGO! to suit your personal requirements. This modular structure, however, has little to no influence on programming. You can still program your LOGO! in the usual way. All you have to take into consideration is that AS-Interfaces and analog I/Os are determined by the position at which the relevant expansion module is installed.

Current devices

Version	Name	Order no.
Standard	LOGO! 12/24RC (DC)	6ED1 052-1MD00-0BA6
Standard	LOGO! 24 (DC)	6ED1 052-1CC00-0BA6
Standard	LOGO! 24RC (AC/DC)	6ED1 052-1HB00-0BA6
Standard	LOGO! 230RC (AC)	6ED1 052-1FB00-0BA6
Standard	LOGO! 12/24RCo (DC)	6ED1 052-2MD00-0BA6
Standard	LOGO! 24RCo (AC/DC)	6ED1 052-2HB00-0BA6
Standard	LOGO! 230RCo (AC)	6ED1 052-2FB00-0BA6
Standard	LOGO! 24o (DC)	6ED1 052-2CC00-0BA6

Memory space is identical for all these devices.

The following expansion modules are available for the standard devices

Version	Name	Order no.
Digital	LOGO! DM8 230R	6ED1 055-1FB00-0BA1
Digital	LOGO! DM16 230R	6ED1 055-1FB10-0BA0
Digital	LOGO! DM8 24	6ED1 055-1CB00-0BA0
Digital	LOGO! DM16 24	6ED1 055-1CB10-0BA0
Digital	LOGO! DM8 12/24R	6ED1 055-1MB00-0BA1
Digital	LOGO! DM8 24R (AC/DC)	6ED1 055-1HB00-0BA0
Digital	LOGO! DM16 24R	6ED1 055-1NB10-0BA0
Analog	LOGO! AM2 (DC)	6ED1 055-1MA00-0BA0
Analog	LOGO! AM2 PT100 12/24	6ED1 055-1MD00-0BA0
Analog	LOGO! AM2 AQ (DC)	6ED1 055-1MM00-0BA1
Text Display	LOGO! TD	6ED1 055-4MH00-0BA0

You can use the description to read off the different properties of LOGO!:

- **12** means 12 V Version
- **24** means 24 V Version
- **230** means 115/230 V Version
- **R** means relay outputs (without R transistor outputs)
- **C** means integrated time/timer switch
- **o** means without display

Current devices

The following memory cards and battery cards are available:

Description	Name	Order no.
Memory card	LOGO! 12/24RC (DC)	6ED1 056-1DA00-0BA0
Battery card	LOGO! 24 (DC)	6ED1 056-6XA00-0BA0
Combined memory and battery card	LOGO! 24RC (AC/DC)	6ED1 056-7DA00-0BA0

Special functions, depending on LOGO! versions

Special function	0BA0 Stan-dard	0BA0 ...L	0BA0 ...LB11	0BA1 all	0BA2 all	0BA3 all	0BA4 all	0BA5 all	0BA6 all
On delay	X	X	X	X	X	X	Xr	Xr	Xr
Off delay	X	X	X	X	X	X	Xr	Xr	Xr
Retentive on delay	X	X	X	X	X	X	Xr	Xr	Xr
On/off delay	-	-	-	-	X	X	Xr	Xr	Xr
Latching relay	X	Xr	Xr	Xr	Xr	Xr	Xr	Xr	Xr
Pulse relay	X	Xr	Xr	Xr	Xr	Xr	Xr	Xr	Xr
Wiping relay	-	X	X	X	X	X	Xr	Xr	Xr
Edge-triggered wiping relay	-	-	-	-	X	X	Xr	Xr	Xr
Weekly timer)*	X	X	X	X	X	X	X	X	X
Yearly timer)*	-	-	-	X	X	X	X	X	X
Up/down counter	X	Xr	Xr	Xr	Xr	Xr	Xr	Xr	Xr
Hours counter	-	Xr	Xr	Xr	Xr	Xr	Xr	Xr	Xr
Symmetrical pulse generator	X	X	X	X	X	X	X	X	X
Asynchronous pulse generator	-	-	-	X	X	X	Xr	Xr	Xr
Random generator	-	-	-	-	X	X	X	X	X
Analog threshold trigger	-	X	X	X	X	X	X	X	X
Analog trigger	-	-	-	-	X	X	X	X	X
Analog comparator	-	-	-	-	X	X	X	X	X
Stairway lighting switch	-	-	-	-	X	X	Xr	Xr	Xr
Multiple function switch	-	-	-	-	X	X	Xr	Xr	Xr
Message text	-	-	-	-	X	X	X	X	X
Softkey	-	-	-	-	-	Xr	Xr	Xr	Xr
Shift register	-	-	-	-	-	-	Xr	Xr	Xr
Analog value monitoring	-	-	-	-	-	-	Xr	Xr	Xr
Analog amplifier	-	-	-	-	-	-	X	X	X
Analog Differential trigger SFBs	-	-	-	-	-	-	X	X	X
Analog multiplexer	-	-	-	-	-	-	-	X	X
Controller	-	-	-	-	-	-	-	Xr	Xr
Ramp control	-	-	-	-	-	-	-	-	X
Pulse width modulator (PWM)	-	-	-	-	-	-	-	-	X
Analog math	-	-	-	-	-	-	-	-	X
Analog math error detection	-	-	-	-	-	-	-	-	X

X = Yes; - = No

)* = Use of this function only makes sense for LOGO! versions with integrated real-time clock.

r= retentive

5.4.3 Memory

5.4.3.1 Memory requirements

The blocks in your circuit program require a certain amount of memory space. The table shows you how much of the memory space each block occupies.

Memory space required for data backup after power failure is specified in the "Rem" column (retentivity enabled).

Block	RAM (Bytes)	Rem (Bytes)
AND (with/without edge evaluation)	12	-
NAND	12	-
OR (with/without edge evaluation)	12	-
NOR	12	-
XOR	8	-
NOT	4	-
On delay	8	3
Off delay	12	3
On/off delay	12	3
Retentive on delay	12	3
Wiping relay	8	3
Edge-triggered wiping relay	16	4
Asynchronous pulse generator	12	3
Random generator	12	-
Stairway lighting switch	12	3
Multiple function switch	16	3
Weekly timer	20	-
Yearly timer	12	-
Up/down counter	28	5
Hours counter	28	9
Analog threshold trigger	16	-
Analog trigger	16	-
Analog differential trigger	16	-
Analog comparator	24	-
Analog value monitoring	20	-
Analog amplifier	12	-
Pulse width modulator (PWM)	24	-
Analog math	20	-
Analog math error detection	12	1
Latching relay	8	1
Pulse relay	12	1
Message text	8	-
Softkey	8	2
Analog multiplexer	20	-
PI Controller	40	2
Ramp control	36	-
Shift register	12	1

0BA0-0BA3: The following specifications apply:

The blocks in your circuit program require a certain amount of memory space. The table below shows you how much memory space the blocks use in the various memory areas:

Block	Par	RAM	Timer	REM
Basic functions	0	0	0	0
On delay	1	1	1	0
Off delay	2	1	1	0
On/off delay	2	1	1	0
Retentive on delay	2	1	1	0
Wiping relay	1	1	1	0
Edge-triggered wiping relay	1	1	1	0
Symmetrical pulse generator	1	1	1	0
Asynchronous pulse generator	3	1	1	0
Random generator	2	1	1	0
Stairway lighting switch	1	1	1	0
Dual-function switch	2	1	1	0
Weekly timer	6	2	0	0
Yearly timer	2	0	0	0
Up/down counter*	2	(2)	0	(2)
Hours counter	2	0	0	4
Threshold trigger	3	3	1	0
Analog trigger	4	2	0	0
Analog comparator	3	4	0	0
Latching relay*	0	(1)	0	(1)
Pulse relay*	0	(1)	0	(1)
Message texts	1	0	0	0
Softkey*	1	(1)	0	(1)

*: Depending on whether the function is configured with or without retentivity, it occupies the following memory space:

- Retentivity off: The function occupies RAM space
- Retentivity on: The function occupies REM space

Here you will find information on memory space provide by LOGO!.

5.4.3.2 Memory space

You may use up to 200 blocks in your circuit program.

The maximum memory space used by a LOGO! circuit program is:

- RAM: 3800 bytes
- Retentive data: 250 bytes

The Info Window displays the memory space used when you call the Tools -> Determine LOGO! function, or when you press the function key [F2].

The following specifications apply

LOGO! series	Blocks	Par	RAM	Timer	REM	Flags
LOGO! 0BA4 ... 0BA5	130	Not restricted	Not restricted	Not restricted	60	24
LOGO! 0BA2 ... 0BA3	56	48	27	16	15	8
LOGO! 0BA1	56	48	27	16	15	4
LOGO! 0BA0	30	27	24	10	0/7	0

Program path

A program path consists of a number of blocks, which start with an input and end with an output.

Nesting depth (LOGO! series 0BA0, 0BA1)

The number of blocks in a program path describes the nesting depth.

I/Os in the sense of nesting depth are:

- Inputs (I, AI)
- High, Low (Hi, Lo)
- Memory markers (M)
- Outputs (Qblock.Output)

Blocks in the sense of nesting depth are:

- Basic functions
- Special functions

Further inputs/outputs according to the nesting depth are:

- Inputs AS-Interface (Ia)
- Outputs AS-Interface (Qa)

LOGO! restricts the nesting depth for your circuit program. Your circuit program has a maximum nesting depth of 58 objects.

This determines the following::

		1 input
	+	56 blocks
	+	1 output
max. depth of the program path::	**=**	**58 objects**

During a simulation or download the system returns an error message if you enter a program path in LOGO!Soft Comfort that exceeds the maximum nesting depth

Implementing longer program paths

Longer program paths can be implemented by means of signal recursion. This is done by appending a flag block to the end of the path. The remaining blocks are connected to the output of the flag block. LOGO! interprets the flag as output of the first program path and also as terminal block of the downstream blocks. LOGO! interprets the long path with flag as two separate paths. If no more flags are available, you can use an output block instead.

5.4.4 Blocks and block numbers

5.4.4.1 Blocks

Blocks represent terminals or functions. LOGO!Soft Comfort distinguishes between various types of block and identifies these by means of an abbreviation.

Block type	Identifier	Block type	Identifier
Input	I	Flag	M
Output	Q	High	Hi
Function	B	Low	Lo

5.4.4.2 Block numbers

Block number assignment

LOGO!Soft Comfort assigns every block you insert in the circuit program a **block number**. LOGO! displays the number of the current block at the top right of the display. LOGO!Soft Comfort displays the block number directly above the inserted block.

Block numbers are used for orientation on the LOGO! display and for the assignment of logical links. In LOGO!Soft Comfort you can also track cut connections by means of their indicated block number.

The corresponding terminal name on the LOGO! or a simple block name replaces the block number at constants and terminals. Each input, output and flag can be assigned further block identifiers via comments. The high and low signal blocks do not have a block number.

Determination of block numbers on a LOGO!

LOGO! has no default position for analog inputs or digital outputs. The respective block number is determined by the hardware structure.

On a LOGO! without modular structure, the position of an analog or digital output is fixed, for example.

6 Tips and Tricks

6.1 Tips and tricks

How to maintain an overview during simulation

A quick and easy way of selecting blocks and placing these into your circuit program

A quick and easy way of connecting blocks in large circuit programs

How to use the Info Window texts for your documentation

A quick and easy way of increasing/reducing the size of the Info Window

How to display the corresponding tooltip for a function key

How to identify your circuit program version

How to access functions via the shortcut menu

A quick and easy way of zooming your circuit program window

A quick way of changing block parameters

A quick way of closing LOGO!Soft without saving the data

How to establish the cycle time

6.1.1 How to maintain an overview during simulation

It may be difficult in simulation mode to maintain a clear overview of large circuit programs and/or when working on low resolution screens. We advise the following procedure:

1. Maximize the LOGO!Soft Comfort application window to full screen size.
2. Close the Info Window and the catalog.
3. Position the mouse pointer onto the small strip, directly at the left side of the icons of the circuit program inputs. Keep the left mouse button pressed and drag and drop the input toolbar out of the LOGO!Soft Comfort application window to the top edge of the screen.
4. Do the same with the toolbar of your circuit program outputs, as described under 2.

Advantage: The space for editing the circuit program has increased. You can still access the I/O toolbars without restriction, since they always remain in the foreground.

Note: You can restore the I/O toolbars to their original position by left-clicking the small cross icon in the upper right corner of the toolbar.

6.1.2 A quick and easy way of selecting blocks and placing these into your circuit program

You have two alternatives to the standard selection of blocks from the programming toolbar icons:

Alternative 1

1. Open the catalog from the programming toolbar.
2. Click on the required block in the catalog to select it.
3. In your circuit program, left-click the block insert position. The block appears at the correct position.
4. To insert further instances of this block, left-click on the relevant insert positions.
5. To insert a further block, select it from the catalog and proceed as described under item 3 and 4.

Advantage: When you change between constants/terminals, basic functions and SFBs, you save yourself having to click the relevant icons in the programming toolbar.

Alternative 2

1. Open the catalog of the programming toolbar.
2. Click on any block in the catalog to select it.
3. If you are creating a large program, you can close the catalog and also hide the programming toolbar.
4. Hold down the Ctrl key and left-click the block insert position in your circuit program. You are displayed a mask with block list, from which you can select the required block with a double-click.
5. Tip: In the mask header, you will also find an input field. You could, for example, enter the initial letter of the required SFB to restrict the display in the mask to a list of blocks with this initial. You thus do not have to browse the entire mask, and you can quickly find the relevant block.
   The block is inserted at the correct position in your circuit program.
6. To insert further instances of this block, left-click on the relevant insert positions.
7. To insert a further block, select it from the catalog and proceed as described previously.

Advantage: You do not have to depend on the catalog and the programming toolbar to create large programs. You can thus close and hide these to provide more screen space for your circuit program.

6.1.3 A quick and easy way of connecting blocks in large circuit programs

In addition to the conventional method of creating connections with the programming toolbar icons, you have another alternative:

1. After you have placed the blocks into the circuit program, double-click on the input or output of a block.
2. A mask opens with a list of the target blocks. Double-click to select a block.
   Tip: In the mask header you will find an input field. You could, for example, enter the initial letter of the required SFB to restrict the display in the mask to a list of blocks with this initial. You thus do not have to browse the entire mask, and you can quickly find the relevant block.
   In addition, you can also use wildcards such as * or ?.
3. The connection is made.

Advantage: Particularly when you are handling large circuit programs, this method provides you with a quick and easy means of creating connections.

6.1.4 How to use the Info Window texts for your documentation

4. Use the mouse to mark the text you want to copy to your documentation.
5. Click the icon  . The selected text is copied to the clipboard of your operating system.
6. Change to your documentation editor.
7. Call the Edit menu and select the Paste command to insert the text from the clipboard into your documentation.

6.1.5 A quick and easy way of increasing/reducing the size of the Info Window

Double-click the title bar of the Info Window to switch the window to full-screen mode in the LOGO!Soft Comfort application window. Double-click the title bar of the Info Window to restore it to its original size.

6.1.6 How to display a corresponding tooltip for a function key

Prerequisite: The tooltips are enabled.

Under Tools -> Options: Look & Feel, select **Metal** or **Extended Windows Look and Feel.**

With the help of the mouse-over-icon function, LOGO!Soft Comfort shows not only the tooltip for this icon, but also the corresponding function key (if available).

Here you will find an overview of the shortcuts.

6.1.7 How to identify your circuit program version

The first 16 characters you enter in the **Project name** field in the File -> Properties: General dialog are downloaded to the LOGO!. The version identifier included with these 16 characters is maintained when you download and upload the circuit program between the PC <-> LOGO!.

This special function is only available with devices as of hardware series 0BA2.

6.1.8 How to access functions via the shortcut menu

Right-click on an object to open a context sensitive window that offers you all the major functions.

6.1.9 A quick and easy way of zooming your circuit program window

Press [CTRL] and turn the mouse wheel.

Result: The size of your circuit program window changes

6.1.10 A quick way of changing block parameters

Click the parameter field you want to change. Press [CTRL] and turn the mouse wheel.

Result: The parameter changes

6.1.11 A quick way of closing LOGO!Soft Comfort without saving the data

Open the **File** menu, press [CTRL] and click the **Close** menu command.

Result: LOGO!Soft Comfort is closed without prompt.

Caution: New or changed circuit programs will not be saved.

6.1.12 How to establish the cycle time

The cycle time is the pure program processing time (reading inputs, executing programs and writing outputs).

The cycle time of each function is less than 0.1 ms. The cycle time of the circuit program can be established using a test program. Refer to the LOGO! manual, appendix B for more information.

With the LOGO! hardware series 0BA3 or older no statements can be made regarding the cycle time of individual functions. The cycle times are different for each function. You can only establish the time for one program cycle. Refer to the LOGO! manual, appendix B for more information. You can download this from the LOGO! homepage on the Internet.

7 Index

D

E

F

G

H

I

L

M

N

O

P

R

S

T

U

V

W

X

Y

Z