Welcome to LOGO!

Dear customer,

Thank you for purchasing LOGO!, and congratulations on your decision. In LOGO! you have acquired a logic module that meets the stringent quality requirements of ISO 9001.

LOGO! is universal in application. Its comprehensive functionality and great ease of use make it a highly cost-efficient solution for virtually any application.

LOGO! documentation

This LOGO! manual tells you how to install, program and use LOGO!.

You can find information on wiring in the LOGO! manual as well as in the LOGO! product information that is supplied with each device. You can get further information on programming LOGO! via the PC in the LOGO!Soft and LOGO!Soft Comfort online help system. LOGO!Soft and LOGO!Soft Comfort are programming software for PCs. They run under WINDOWS$^{\circledR}$ and will help you get to know LOGO! and to write, test, print out and archive programs independently of LOGO!.

Guide to the manual

We have subdivided this manual into 8 chapters:
- Getting to know LOGO!
- Installing and wiring LOGO!
- Programming LOGO!
- Parameterizing LOGO!
- LOGO!'s program modules
- LOGO! software
- Applications
- Appendices (technical specifications, determination of memory requirements, LOGO! without display, AS interface-specific information, order numbers)

Additional support

You can find answers to your LOGO! questions quickly and easily on the Internet at http://www.ad.siemens.de/logo .

Safety guidelines

This manual contains notices which you should observe to ensure your own personal safety, as well as to protect the product and connected equipment. These notices are highlighted in the manual by a warning triangle and are marked as follows according to the level of danger:

Danger

Indicates that death, severe personal injury or substantial property damage **will** result if proper precautions are not taken.

Warning

Indicates that death, severe personal injury or substantial property damage **can** result if proper precautions are not taken.

Caution

Indicates that minor personal injury or property damage can result if proper precautions are not taken.

Note

Draws your intention to particularly important information on the product, handling the product, or to a particular part of the documentation.

Warning

Only **qualified personnel** should be allowed to install and work on this equipment. Qualified personnel are defined as persons who are authorized to commission, to ground and to tag circuits, equipment and systems in accordance with established safety practices and standards.

Warning

This device may only be used for the applications described in the catalog and the technical description, and only with non-Siemens devices or components if they have been approved or recommended by Siemens.
This product can only function correctly and safely if it is transported, stored, set up, and installed correctly, and operated and maintained as recommended.

Disclaimer of liability

We have checked the contents of this manual for agreement with the hardware and software described. Since deviations cannot be precluded entirely, we cannot guarantee full agreement. However, the data in this manual is reviewed regularly and any necessary corrections included in subsequent editions. Suggestions for improvement are welcomed.

Contents

Contents

1 Getting to know LOGO!

What is LOGO! ?

LOGO! is the universal logic module from Siemens.

LOGO! integrates

- Control functions
- An operating and display unit
- Power supply
- An interface for program modules and a PC cable
- Ready-to-use basic functions that are often required in day-to-day operation, such as functions for on/off delays and current impulse relays
- Time switch
- Binary markers
- Inputs and outputs according to the device type

You can use LOGO! for domestic and installation engineering tasks (e.g. stairway lighting, external lighting, sun blinds, shutters, shop window lighting etc), switch cabinet engineering and mechanical and apparatus engineering (e.g. gate control systems, ventilation systems, or rainwater pumps etc).

You can also use LOGO! for specific control systems in conservatories or greenhouses, for signal preparation in control systems and, with the ASi variant, you can have on-site centralized control of machines and processes.

There are special variants without an operating unit for serial applications in the construction of small machines, apparatus and control cabinets and the field of installation.

What devices are available?

LOGO! is available with 12 V, 24 V and 230 V for the following variants:

- Standard variant with 6 inputs and 4 outputs with dimensions of 72 x 90 x 55 mm
- Variant without display with 6 inputs and 4 outputs with dimensions of 72 x 90 x 55 mm
- Long variant with 12 inputs and 8 outputs with dimensions of 126 x 90 x 55 mm
- Bus variant with 12 inputs and 8 outputs, and additional AS interface bus connection, via which 4 further inputs and 4 further outputs are available in the bus system. And all this packed into dimensions of 126 x 90 x 55 mm.

All variants include 19 ready-to-use basic and special functions for program creation.

It's your choice

The various variants can be adapted very easily to your own specific task.

LOGO! provides solutions ranging from the small domestic installation through small automation tasks to extensive tasks integrating the AS interface bus system.

How LOGO! is structured

LOGO!...standard variant

LOGO!...L/LOGO!...LB11

① Power supply

② Inputs

③ Outputs
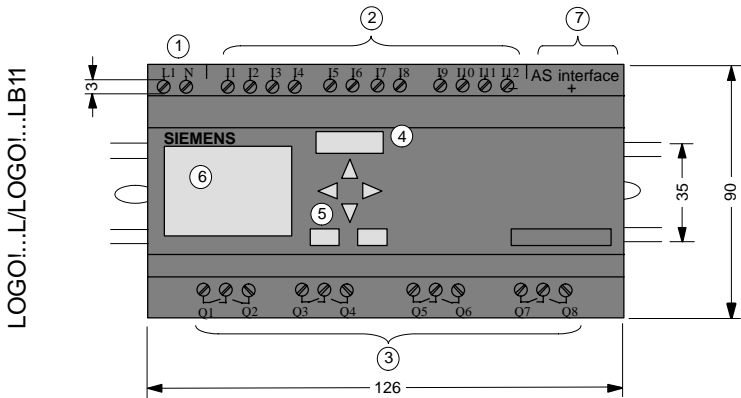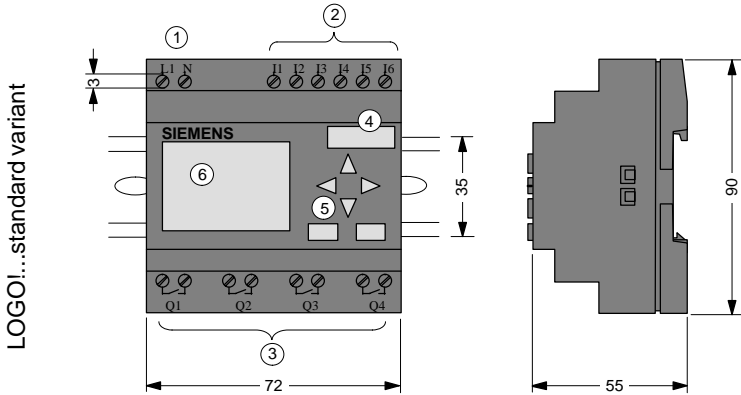
④ Module shaft with
cover

⑤ Control panel
(not with RCo)

⑥ LCD
(not with RCo)

⑦ AS interface connection
(only with LB11)

How to recognize which LOGO! variant you have

LOGO!'s designation contains information on various characteristics:

- 12: 12 V DC variant
- 24: 24 V DC variant
- 230: 115/230 V AC variant
- R: Relay outputs (without R: Transistor output)
- C: Integrated seven-day time switch
- o: Variant without display
- L: Twice the number of outputs and inputs
- B11: slave with AS interface bus connection

We also use icons to describe the different types of LOGO! They are used whenever information refers to only some of the LOGO! variants:

Standard variant with 6 inputs and 4 outputs with dimensions of 72 x 90 x 55 mm

Variant without display with 6 inputs and 4 outputs with dimensions of 72 x 90 x 55 mm

..L variant with 12 inputs and 8 outputs with dimensions of 126 x 90 x 55 mm

..B11 variant with 12 inputs and 8 outputs and additional AS interface bus connection with 4 virtual inputs and 4 virtual outputs with dimensions of 126 x 90 x 55 mm.

Variants

The following different variants of LOGO! are available:

Symbol	Designation	Outputs	Type
	LOGO! 12RC	4 * 230 V; 10A	Relay
	LOGO! 24	4 * 24 V; 0.3 A	Transistor
	LOGO! 24RC	4 * 230 V; 10A	Relay
	LOGO! 230RC	4 * 230 V; 10A	Relay
	LOGO! 24RCo	4 * 230 V; 10A	Relay
	LOGO! 230RCo	4 * 230 V; 10A	Relay
	LOGO! 12RCL	8 * 230 V; 10A	Relay
	LOGO! 24L	8 * 24 V; 0.3 A	Transistor
	LOGO! 24RCL	8 * 230 V; 10A	Relay
	LOGO! 230RCL	8 * 230 V; 10A	Relay
	LOGO! 24RCLB11	8 * 230 V; 10A	Relay
	LOGO! 230RCLB11	8 * 230 V; 10A	Relay

Certification, recognition and approval

LOGO! has to UL, CSA and FM certification.

- UL listing mark
  Underwriters Laboratories (UL) to
  UL 508 standard , file no. 116536
- CSA–Certification–Mark
  Canadian Standard Association (CSA) to
  standard C22.2 No. 142, file no. LR 48323
- FM approval
  Factory Mutual (FM) Approval to
  Standard Class Number 3611, Class I, Division 2,
  Group A, B, C, D

⚠️ **Warning**

Personal injury and material damage may be incurred.

In potentially explosive areas, personal injury or property damage can result if you withdraw any connectors while the system is in operation.

Always ensure that the system is deenergized before you disconnect LOGO! plug connections and associated components in potentially explosive areas.

LOGO! carries CE marking, complies with the VDE 0631 and IEC1131 standards and has interference suppression to EN 55011 (limit class B, class A for ASi bus operation).

Shipbuilding certification has been granted.

- ABS – American Bureau of Shipping
- BV – Bureau Veritas
- DNV – Det Norske Veritas
- GL – Germanischer Lloyd
- LRS – Lloyds Register of Shipping
- PRS – Polski Rejestr Statków

LOGO! can therefore be used both in industry and at home.

2 Installing and wiring LOGO!

General

We will show you how to install and deinstall LOGO! with the aid of an illustration of the LOGO! 230RC. The measures described also apply to all other LOGO! modules.

Install LOGO! in a distribution box or control cabinet, ensuring that the connectors are covered. If they are not, there is a danger of touching live parts.

LOGO! must be installed and wired by a trained technician who knows and complies with both the universally applicable engineering rules and the regulations and standards that apply in specific cases.

Dimensions

The dimensions of LOGO! comply with DIN 43880.

LOGO! must be snapped onto a DIN rail with a width of 35 mm to DIN EN 50022.

Width of LOGO!:

- LOGO! is 72 mm wide, which corresponds to the size of 4 modules (standard variant).
- LOGO!...RCo is 72 mm wide, which corresponds to the size of 4 modules (standard variant).
- LOGO!...L is 126 mm wide, which corresponds to the size of 7 modules.
- LOGO!...B11 is 126 mm wide, which corresponds to the size of 7 modules.

2.1 Installing/deinstallingLOGO!

Installing

To **install** LOGO! on a DIN rail, proceed as follows:

1. Place LOGO! on the rail
2. Swivel it onto the rail so that the snap catch on the back of LOGO! engages.

Depending on the type of DIN rail used, the snapping mechanism may be a bit stiff. If it is too stiff and LOGO! won't snap on, you can pull the snap catch down a little as you do when deinstalling LOGO! (as described below).

Deinstalling

To **deinstall** LOGO!, proceed as follows:

1. Insert a screwdriver in the hole shown in the picture at the lower end of the snap catch, and pull the snap catch downwards.
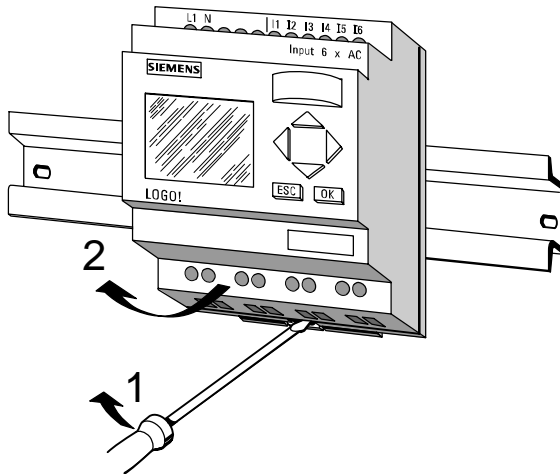
2. Swivel it away from the DIN rail

2.2 Wiring LOGO!

Use a screwdriver with a 3 mm head to wire LOGO!.

You don't need wire end ferrules for the connectors. You can use wires up to the following sizes:

- 1 x 2.5 mm^2
- 2 x 1.5 mm^2 for each second connector compartment

Connecting torque: 0.4...0.5 Nm or 3...4 LBin

2.2.1 Connecting the power supply

LOGO! 230 variants are suitable for line voltages with a rating of 115 V AC and 230 V AC and LOGO! 24 variants and 12 variants are suitable for 24 V DC and 12 V DC sup-ply voltage. LOGO! 24 RC/RCo is also suitable for a supply voltage of 24 V AC. Please note the technical specifica-tions in Appendix A that refer to the permissible voltage tolerances, mains frequencies and current consumption.

Note

Power failure might result for instance in an additional edge after power restoration with edge-triggered special functions.

Connecting

To connect LOGO! to the system, proceed as follows:

LOGO! 12/24...   LOGO! 230...
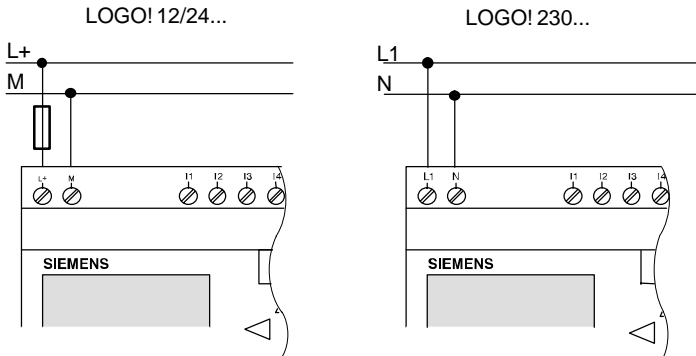
Protection by fuse
if required (recommended).

12/24 RC...:	0.8 A
24:	2.0 A
24 L:	3.0 A

Note

LOGO! has protective insulation. A ground terminal is not necessary.

2.2.2   Connecting LOGO!'s inputs

Prerequisites

Connect sensors to the inputs. The sensors may be push-buttons, switches, photoelectric barriers, daylight control switches etc.

Sensor attributes for LOGO!

	LOGO! 12 RC	**LOGO! 24**	**LOGO! 24 RC/RCo**	**LOGO! 230 RC/RCo**
Circuit state 0	< 4 V DC	< 5 V DC	< 5 V AC/DC	< 40 V AC
Input current	< 0.5 mA	< 0.5 mA	< 1.0 mA	< 0.03 mA
Circuit state 1	>8 V DC	>12 V DC	> 12 V AC/DC	> 79 V AC
Input current	> 1.5 mA	> 1.5 mA	> 2.5 mA	> 0.08 mA

	LOGO! 12 RCL	**LOGO! 24 L**	**LOGO! 24 RCL...**	**LOGO! 230 RCL...**
Circuit state 0	< 4 V DC	< 5 V DC	< 5 V DC	< 40 V AC
Input current	< 0.5 mA	< 1.5 mA	< 1.5 mA	< 0.03 mA
Circuit state 1	>8 V DC	>12 V DC	> 12 V DC	> 79 V AC
Input current	> 1.5 mA	> 4.5 mA	> 4.5 mA	> 0.08 mA
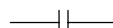
Sensor connections

Connecting glow lamps, 2-wire proximity switch for LOGO! 230RC/230RCo
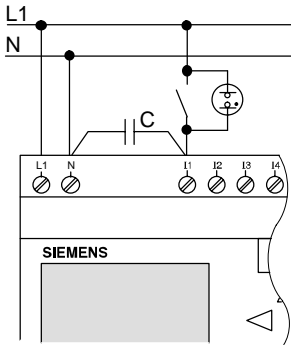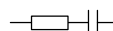
Order number for C:

Siemens
Switching Devices & Systems

3SB1430–3C
3SB1420–3D

3TX7462–3T

Circuit state change 0 ⋮ 1 / 1 ⋮ 0

When the circuit state changes from 0 to 1, circuit state 1 and, in the case of a change from 1 to 0, circuit state 0 must be in place for at least one program cycle for LOGO! to recognize the new circuit state.

The cycle time of the program processing depends on the size of the program.

In the appendix you can find a description of a short test program that will help you to work out the current cycle time.

Fast inputs

LOGO! (apart from LOGO! 230..., 24 RC and 24 RCo) also has inputs for frequency functions. The same restrictions do not apply to these fast inputs. The fast inputs are the last two inputs on a LOGO!:

- LOGO! standard variant: inputs I5/I6
- LOGO!...L variant: inputs I11/I12

Connecting

To connect the sensors to LOGO!, proceed as follows:
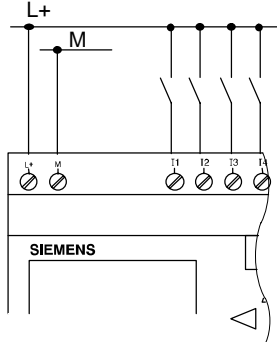
LOGO! 12/24...

The inputs of LOGO! 12/24... are non-isolated and therefore require the same reference potential (ground) as the power supply.

LOGO! 230 ... (standard variant)
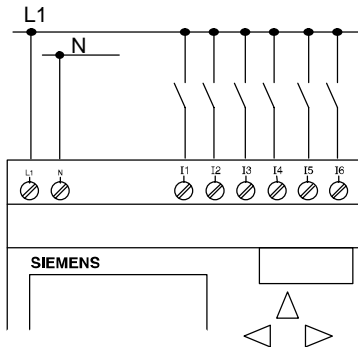
<table>
<tr><td>⚠</td><td>

Warning

Existing safety regulations (VDE 0110, ... and IEC 1131, ..., as well as UL and CSA) prohibit the connection of different phases to the inputs of LOGO! 230.

</td></tr>
</table>

LOGO! ...L...
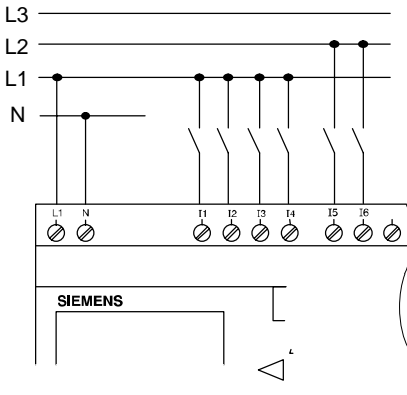
The inputs of LOGO! ...L.. are grouped in groups of 4 inputs. The same applies to these groups as for the individual inputs of a standard LOGO!. Different phases are possible only between the blocks.

⚠ **Warning**

Existing safety regulations (VDE 0110, ... and IEC 1131, ..., as well as UL and CSA) prohibit the connection of different phases to one input block of LOGO! ...**L**...

2.2.3　Connecting outputs

LOGO! ...R...

The outputs of LOGO! ...**R**... are relays. The contacts of the relays are isolated from the power supply and from the inputs.

Prerequisites for the relay outputs

You can connect different loads to the outputs such as lamps, fluorescent tubes, motors, contactors etc. The loads connected to LOGO! ...R... must have the following properties:

- The maximum switched current depends on the type of load and the number of operations. You will find more information on this in the technical specifications.
- When switched on (Q = 1), the maximum current is 10 amperes (8 A at 230 V AC) for a non-inductive load and 3 amperes (2 A at12/24 V AC/DC) for an inductive load.

Connecting

To connect the load to LOGO! ...**R**...variants, proceed as follows:
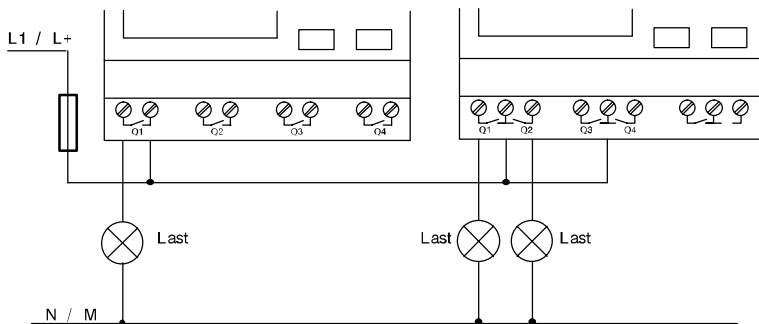
Protection with automatic circuit breaker (max. 16 A, B16, e.g. power circuit breaker 5SX2 116-6 (if desired)

LOGO! with transistor outputs

LOGO! variants with transistor outputs can be identified by the fact that the letter **R** is missing from their type designation. The outputs are short circuit-proof and overload-proof. A separate voltage supply to the load is not necessary since LOGO! supplies the load with voltage.

Prerequisites for transistor outputs

The load connected to LOGO! must have the following properties:

- The maximum switched current is 0.3 amperes per output.

Connecting

To connect the load to LOGO! with transistor outputs, proceed as follows:
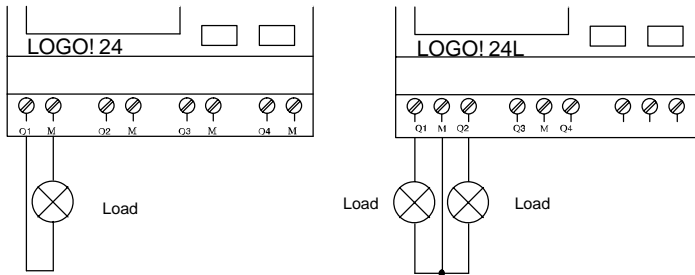
Load: 24 V DC, 0.3 A max.

2.2.4   Connecting the ASi bus (LOGO! ...B11 only)

This section will be of interest to you if you want to connect LOGO!..B11 to the ASi bus.
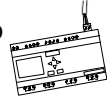
LOGO!...B11

LOGO!...B11 can be integrated into a network as an ASi slave. Using a two-wire lead you can then do the following:

• Read in and process 4 additional inputs via the ASi bus.

• Operate 4 additional outputs on one overlaid master of the ASi bus.

Configure LOGO!...B11 in the ASi bus by means of the ASi master you are using.

Prerequisites for operation on an ASi master

Please note: LOGO! ...B11 must be registered in the ASi system, i. e. LOGO! is assigned an address by the bus master. Please read Section 2.2.5 to find out how to do this with LOGO!.

⚠ **Caution**

The ASI address can be changed at least 10 times for all LOGO! ..LB11 variants.

Additional changes are not guaranteed.

Connecting

Connect the bus connector cable to an approved connector in the system. Make sure that the polarity is correct.

Then push the wired connector into the interface marked **AS interface**.
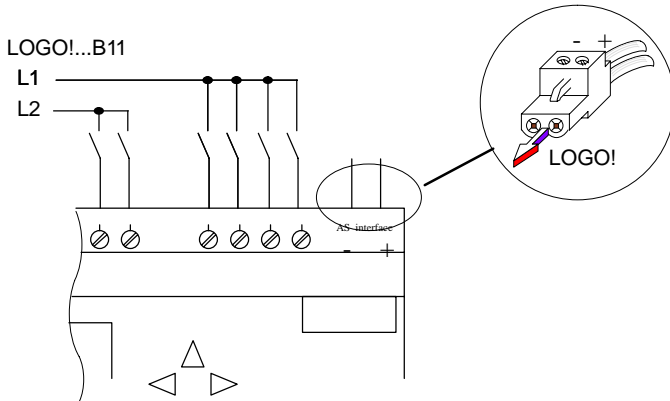
2.2.5   LOGO!...B11 on the ASi bus

LOGO!...B11 must be known to the bus master if you are to be able to use the ASi functionality. This takes places automatically when you connect LOGO!...B11 to the bus lead. The master detects the address of the slave.

In the case of LOGO!...B1, the address preset at the factory = 0. The master assigns a new address that is not equal to 0.

If there are no address conflicts in the system or if only one slave with the address 0 is connected, you do not have to take any further steps.

Note

If you are connecting several slave assemblies (e.g. LOGO!...B11) to the bus at the same time, please read Appendix E.

2.3 Switching LOGO! on/resumption of power

LOGO! does not have a power switch. How LOGO! responds when switched on, depends on the following:

• Whether a program is stored in LOGO!.
• Whether a program module is connected.
• Whether it is a LOGO! variant without a display (LOGO!...RCo).
• The state LOGO! was in before power off.

The table indicates LOGO!'s response to the possible situations:

Before power off

After power on

```
No Program
```
No program in memory

or

(empty)

```
No Program
```

(with program)

```
>Program..
PC/Card..
Start
```

```
I:123456
 Mo 09:00

Q:1234 RUN
```

```
B03:Par
Par = 0300

Cnt = 0028
```
LOGO! in RUN

or

(empty)

```
I:123456
 Mo 09:00

Q:1234 RUN
```
With stored program from LOGO!

(with program)

```
I:123456
 Mo 09:00

Q:1234 RUN
```
With program copied from module to LOGO!

```
B01
&
Q1
```
Program in memory

or

(empty)

```
>Program..
PC/Card..
Start
```
With stored program from LOGO!

(with program)

```
>Program..
PC/Card..
Start
```
With program copied from module to LOGO!

Try to remember the 4 simple rules for starting LOGO!:

1. If there is no program in LOGO! or on the connected program module, LOGO! (with display) displays the message: No Program.

2. If there is a program on the program module, it is automatically copied to LOGO!. If there is already a program in LOGO!, it is overwritten

3. If there is a program in LOGO! or on the program module, LOGO! adopts the operating status it had before power off. If you are using a variant without display (LOGO! ...RCo), it automatically changs from STOP to RUN (the LED changes from red to green).

4. If you have switched on retentivity for at least one function or are using a function with retentivity permanently switched on, its current values are retained at power off.

Note

If a power failure occurs while you are entering a program, you will find when the power is restored that LOGO! no longer contains the program.

You should therefore back up your original program on a program module (card) or on a computer (LOGO!Soft, LOGO!Soft Comfort) before changing it.

LOGO! operating statuses

LOGO! has 2 operating statuses: STOP and RUN

STOP	RUN
• Display: 'No Program' (not LOGO! ...RCo) • Switch LOGO! to programming mode • The LED lights up red (LOGO! ...RCo only)	• Display: Mask for monitoring the inputs and outputs (after START in the main menu) (not LOGO! ...RCo) • Switch LOGO! to parameterization mode (not LOGO! ...RCo) • The LED lights up green (LOGO! ...RCo only)
Action by LOGO!: • The inputs are not read. • The program is not executed. • The relay contacts are always open or the transistor outputs are switched off.	Action by LOGO!: • LOGO! reads the state of the inputs. • LOGO! calculates the state of the outputs with the program. • LOGO! switches the relays/ transistor outputs on or off.

3 Programming LOGO!

The first steps with LOGO!

By programming we mean entering a circuit. A LOGO! programm is really no more than a circuit diagram represented in a different way.

We have changed the way it is represented to suit LOGO!'s display panel. In this chapter we will show you how to use LOGO! to turn your applications into LOGO! programs.

Note

The LOGO! variants without a display, LOGO! 24RCo and LOGO! 230RCo, don't have an operating unit. They are mainly intended for serial applications in small machine and apparatus construction.

LOGO!...RCo variants are not programmed on the device. Programs from LOGO! software or from memory modules of other LOGO! devices are transferred into the device.

In the first section of the chapter a brief example will help you get to know how to use LOGO!.

- First of all we will begin by introducing the two basic terms **connector** and **block**, and show you what is meant by these terms.
- In a second step we will develop a program from a simple, conventional circuit.
- In the third step you can then enter this program directly into LOGO!.

After reading through only the first few pages of this manual, you will already have stored your first executable program in LOGO!. Using suitable hardware (switches etc) you will then be able to carry out your first tests.

What else can you look forward to?

As you progress further through the chapter, you will expand your first program and learn a number of techniques you can use to make changes to an existing program.

Finally in the third and last section of the chapter we will introduce you to LOGO! in its entirety. This covers:

- All the functions of LOGO!
- Introduction to the menu structure of LOGO!

3.1 Connectors

LOGO! has inputs and outputs:
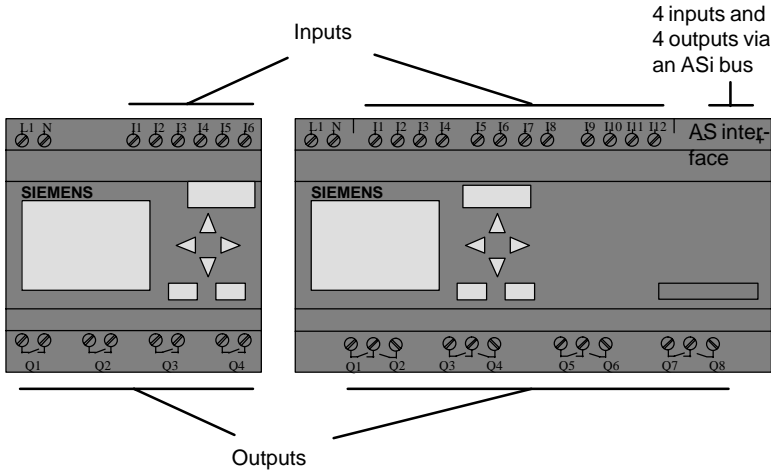
Each input is identified by the letter I and a number. When you look at LOGO! from the front, you can see the connectors for the inputs at the top.

Each output is identified by the letter Q and a number. You can see the connectors of the outputs in the figure below.

Note

Inputs and outputs that are made available with LOGO!...B11 by means of the AS interface bus connection are physical inputs on LOGO! itself.

Note that it is the bus master that defines the input and output devices on the ASi bus.

LOGO!'s connectors
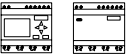
The term connector refers to all connections and states in LOGO!.

The inputs and outputs can have the state '0' or '1'. '0' means there is no voltage at the input, '1' means that there is. But that is unlikely to be new to you.

We introduced the connectors hi, lo and x, in order to facilitate program entry for you. 'hi' (high) has the fixed state '1' and 'lo' (low) has the fixed state '0'.

If you don't want to wire an input on a block, use the 'x' connector. You can find out what a block is on the next page.

LOGO! recognizes the following connectors:

Connec-tors			
Inputs	I1 to I6	I1 to I12	I1 to I12 and Ia1 to Ia4 (AS interface)
Outputs	Q1 to Q4	Q1 to Q8	Q1 to Q8 and Qa1 to Qa4 (AS interface)
lo	Signal with level '0' (off)		
hi	Signal with level '1' (on)		
x	An existing connection that is not used		

3.2 Blocks and block numbers

In this chapter, we will describe how you can create extensive circuits with the aid of LOGO!'s elements and how the blocks are linked to each other and to the inputs and outputs.

For this purpose, please turn to Section 3.3. There we describe how to turn a conventional circuit into a LOGO! program.

Blocks

A block in LOGO! is a function which converts input information into output information. With earlier variants of LOGO! you had to wire up the individual elements in the control cabinet or terminal box.

When you program LOGO! you connect connectors with blocks. To do this, simply select the connection you require from the **Co** menu (Co stands for connector).

Logic operations

The simplest blocks are logic operations:

• AND

• OR

• ...
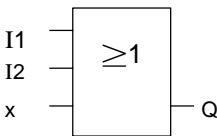
Inputs I1 and I2 are connected to the OR block. The last input of the block is not used and is therefore marked with an x.

We have made the special functions far more powerful than before:

• Current impulse relay

• Counter

• On-delay

• ....

You can find a complete list of all the functions of LOGO! in Section 3.7.

Displaying a block in LOGO!'s display

The figure below shows a typical LOGO! display. As you can see only block can be shown at a time. We have therefore introduced block numbers to help you to check how the circuit is structured.

Display image of LOGO!

Block number – assigned by LOGO!

This shows that a further block is connected

Input

Connection is not required   Block   Output
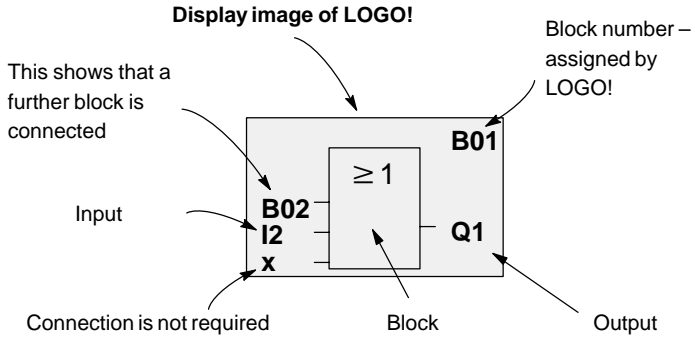
Assigning a block number

Whenever you insert a block in a program, LOGO! assigns that block a number.

LOGO! uses the block number to indicate the connections between the blocks. The block numbers are therefore chiefly meant to help you find your way around the program.

Block number

There is a connection between these blocks

Move around the program using the key◄
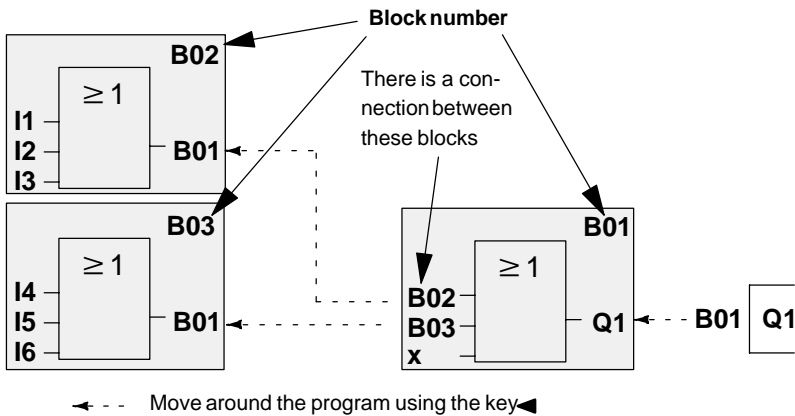
The overview display shows you three displays of LOGO!, which together make up the program. As you can see LOGO! links the blocks with one another by means of the block numbers.

Advantages of the block numbers

You can connect almost any block to an input of the current block using its block number. In this way you can use the interim results of logic or other operations more than once. This saves you the work required to enter things again as well as memory space, and ensures that your circuit remains clear. In this case you have to know how LOGO! has named the blocks.

Note

To make working with LOGO! particularly efficient, we recommend that you draw up a diagram overview of the program. This will make it easier to create the program. You can then enter the block numbers assigned by LOGO! in this diagram.

If you use the LOGO!Soft software to program LOGO!, you can display and print out a ladder program. You can create a functional block diagram of your program straight away using LOGO!Soft Comfort.

3.3 From circuit diagram to LOGO!

How a circuit is represented in a circuit diagram

You know, of course, how a circuit is represented in a circuit diagram. Here is an example:
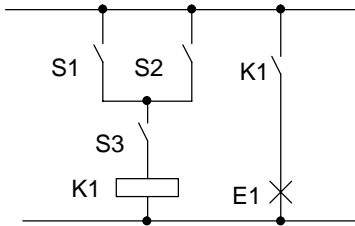
The consumer E1 is switched on and off by means of the switches (S1 **OR** S2) **AND** S3.

The relay K1 picks up, when S1 or S2 and also S3 are closed.

Implementing a circuit with LOGO!

You create a circuit In LOGO! by connecting blocks and connectors to each other:
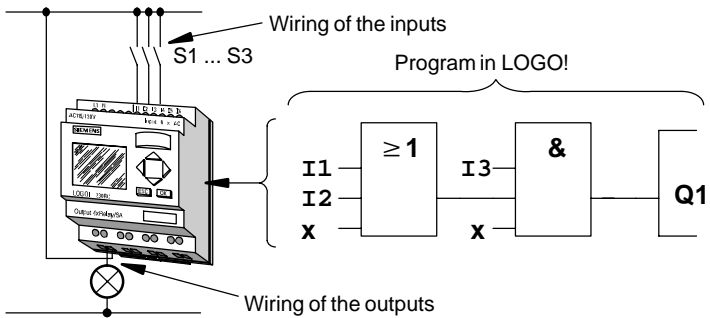
To implement a circuit in LOGO!, begin at the output of the circuit.

The output is the load or the relay that is supposed to operate.

You convert the circuit to blocks. To do this, go through the circuit from the output to the input:

<u>Step 1</u>: At output Q1 there is a series connection of the normally open contact S3 with another circuit component. The series connection corresponds to an AND block:
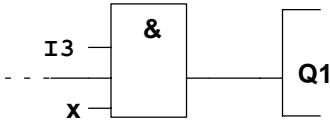
<u>Step 2</u>: S1 and S2 are connected in parallel. The parallel connection corresponds to an OR block:
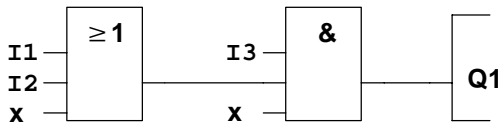
You have now provided a complete description of the circuit for LOGO!. You now need to connect the inputs and outputs to LOGO!.

Wiring

Connect switches S1 to S3 to the screw connectors of LOGO!:

- Connect S1 to connector I1 on LOGO!
- Connect S2 to connector I2 on LOGO!
- Connect S3 to connector I3 on LOGO!

Only 2 inputs of the OR block are used so the third input must be marked as unused. This is indicated by the **x** next to it.

Likewise, only 2 inputs of the AND block are used. The third input is therefore also marked as 'unused' by an **x** next to it.

The output of the AND block controls the relay at output Q1. Consumer E1 is connected at output Q1.

Wiring example

The following table shows you the wiring based on a 230 V variant of LOGO!.

Wiring of the inputs

Wiring of the outputs
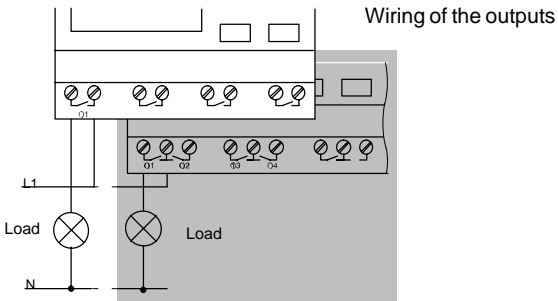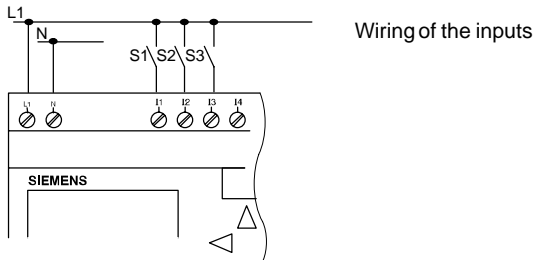
Wiring of the outputs of a LOGO...L...

3.4 The 4 golden rules for working with LOGO!

Rule 1
3-finger grip

- Enter the circuit in programming mode. Switch to programming mode by pressing the 3 keys ◄, ► and **OK** at the same time.
- Change the values of times and parameters in parameterization mode. Switch to parameterization mode by pressing the 2 keys **ESC** and **OK** at the same time.

Rule 2
Outputs and inputs

- Always enter a circuit from output to input.
- You can connect an output to several inputs, but not several outputs to one input.
- You can't connect an output to a preceding input within a program path. Insert markers or outputs in such cases (recursions).

Rule 3
Cursor and cursor movement

The following applies when entering a circuit:

- When the cursor appears in the form of an underscore, you can **move the cursor**:
  - Use the keys ◄, ►, ▼ or ▲ to move the cursor in the circuit.
  - Press OK to select a connector/block.
  - Press ESC to exit circuit input.
- When the cursor appears in the form of a solid block, you should **select a connector/block**
  - Use the keys ▼ or ▲ to select a connector/block.
  - Press OK to accept a selection.
  - Press ESC to go back one step.

Rule 4
Planning

- Before you enter a circuit, draw up a complete plan of it on paper or program LOGO! directly using LOGO!Soft or LOGO!Soft Comfort.
- LOGO! can only store complete programs. If you enter an incomplete program, LOGO! is not able to exit **Programming** mode.

3.5 Overview of LOGO!'s menus

Programming mode

Main menu

```
>Program..
 PC/Card..
 Start
```

OK →

← ESC

Programming menu

```
>Edit Prg
 Clear Prg
 Set Clock
 ASi_BUS..
```

OK

ESC

PC/card menu

```
>PC↔LOGO
 LOGO→Card
 Card→LOGO
```
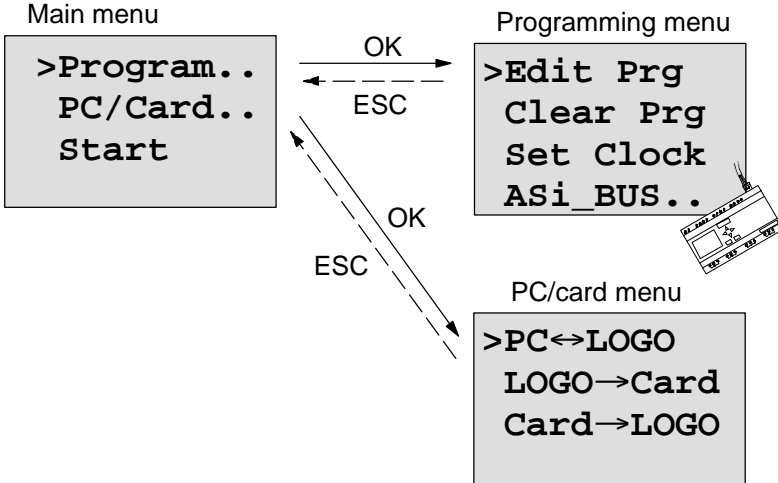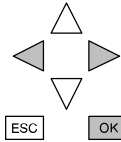
Parameterization mode

Parameterization menu

```
>Set Clock
 Set Param
```
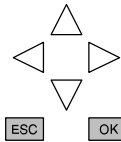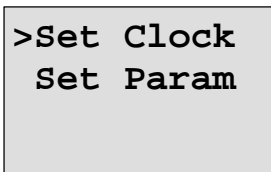
3.6 Entering and starting a program

You have designed a circuit and now want to enter it in
LOGO!. The example below illustrates how to do this.

3.6.1 Switching to programming mode

Connect LOGO! to the mains and switch on the power. The
following message appears on the display:

```
No Program
```
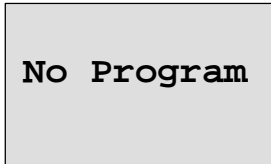
Switch LOGO! to programming mode To do this, press the
keys ◄, ► and **OK** at the same time. The fact that you have
to press the keys simultaneously prevents anyone pressing
them and switching to programming mode inadvertently.

```
No Program
```
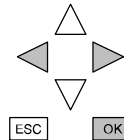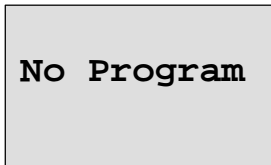
When you press the keys, LOGO!'s main menu appears:

```
>Program..
 PC/Card..
 Start
```
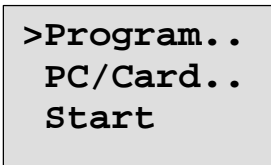
LOGO!'s main menu

On the left in the first line you will see ">". Use the keys ▲ and ▼ to move the ">" up and down. Move the ">" to "Program.." and press **OK**. LOGO! switches to the programming menu.

```
>Edit Prg
 Clear Prg
 Set Clock
 ASi_BUS..
```
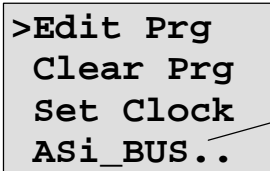
LOGO!'s programming menu

The ASi bus entry only appears with LOGO!...LB11 variants

Here too, you can move the ">" by pressing the ▲ and ▼ keys. Position the ">" on "Edit Prg" (i.e. to enter the program) and press **OK**. LOGO! then displays the first output:
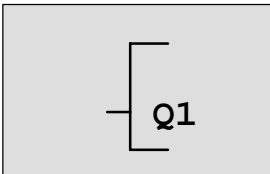
LOGO!'s first output

Use the ▲ and ▼ keys to select the other outputs. At this point, you begin to enter your circuit.

3.6.2 First program

Let's have a look at the following circuit: a parallel connection of two switches.

Circuit diagram

How a circuit is represented in a circuit diagram
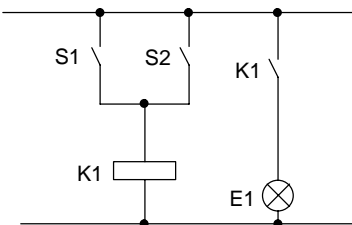
The consumer is switched on by switch S1 or switch S2. As far as LOGO! is concerned, the parallel connection of the switches is an OR block because S1 **or** S2 switches the output on.

Translated into the LOGO! program, this means: Relay K1 (in LOGO! via output Q1) is controlled by an OR block.

Program

I1 and I2 are connected to the input of the OR block, S1 to I1 and S2 to I2.

Thus, the program in LOGO! looks like this:
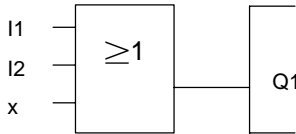
Wiring

The wiring is as follows:
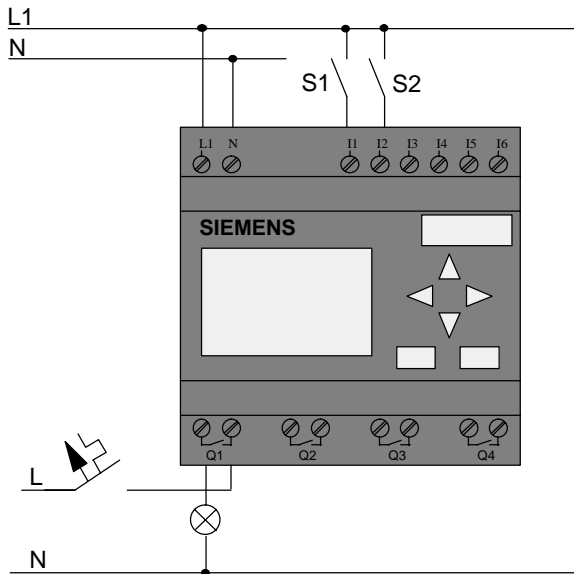
Switch S1 acts on input I1 and switch S2 on input I2. The consumer is connected to relay Q1.

3.6.3 Entering the program

Let's enter the program now (from the output to the input). Initially, LOGO! displays the output:
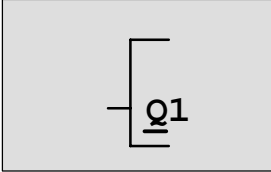
LOGO!'s first output

The Q of Q1 is underlined. This underlining is the **cursor**. The cursor indicates your current position in the program. You can move the cursor by pressing the ▲, ▼, ◄ and ► keys. Now press the key ◄. The cursor moves to the left.
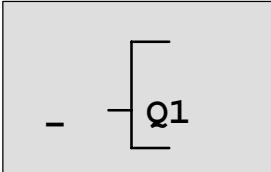
The cursor indicates your position in the program.

At this point, enter only the first block (the OR block). Press **OK** to switch to input mode.
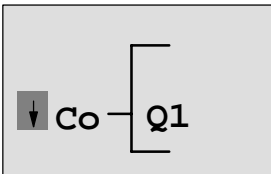
The cursor appears in the form of a solid block: you can select a connector or block.

The cursor no longer appears in the form of an underline; instead it appears as a solid block that flashes on and off. At the same time, LOGO! offers you the first list for selection.

Select the BF list (by pressing ▼ until BF appears) and press **OK**. LOGO! then displays the first block in the list of basic functions (BF):
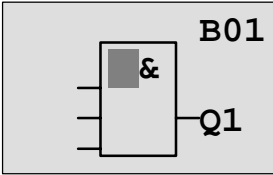
The first block in the list of basic functions is AND. The cursor appears in the form of a solid block, indicating that you have to select a block.

Now press the key ▲ or ▼, until the OR block appears in the display:
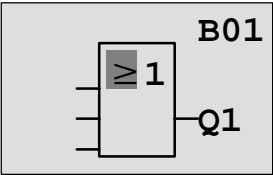
The cursor is still in the block and appears in the form of a solid block.

Press **OK** to conclude your selection.

The following appears in the display panel
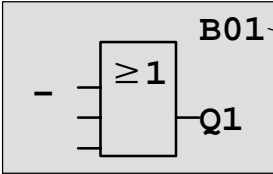
Block number

Your entire program looks like this
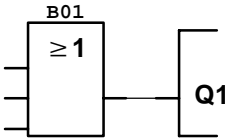
You have now entered the first block. Each block you enter receives a number, the block number. All you have to do now is wire the inputs of the block. To do this, proceed as follows:

Press **OK**.

The following appears in the display panel
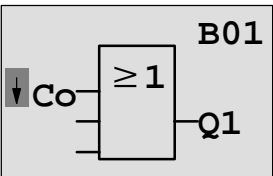
Select the Co list: Press **OK**

The following appears in the display panel
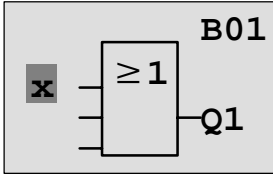
The first item in the Co list is the character for indicating
that the input is not use, an "x". Use the ▲ or ▼ key to se-
lect input I1.
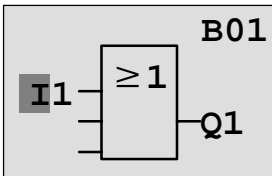
Press **OK**: I1 is connected to the input of the OR block.
The cursor jumps to the next input of the OR block.
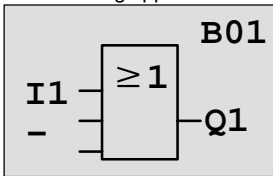
The following appears in the display panel   Your entire program so far
                                                  looks like this in LOGO!
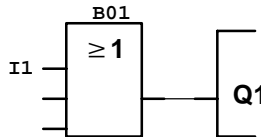
Now connect input I2 to the input of the OR block. You
know how to do this already:

1. Switch to input mode: **OK**
2. Select the Co list: ▲ or ▼
3. Accept the Co list: **OK**
4. Select I2:   ▲ or ▼
5. Accept I2:   **OK**

Thus, I2 is now connected to the input of the OR block:

The following appears in the display panel
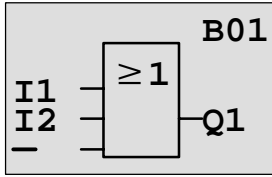
Your entire program so far looks like this in LOGO!
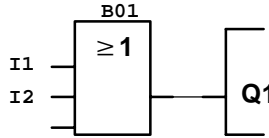
We don't need the last input of the OR block in this program. In a LOGO! program, you mark an input that is not used with an "x", so now enter the 'x' (you know the principle already):

1. Switch to input mode: **OK**
2. Select the Co list: **▲** or **▼**
3. Accept the Co list: **OK**
4. Select x: **▲** or **▼**
5. Accept x: **OK**

Thus, all the block's inputs are now wired. As far as LOGO! is concerned, the program is now complete. LOGO! returns to output Q1.

The following appears in the display panel
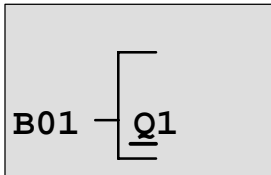
Your program looks like this
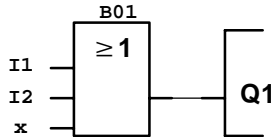
If you want to have another look at your first program, you can use the ◄ or ► key to move the cursor through the program.

But we are going to exit program input now. To do this, proceed as follows:

1. Return to the programming menu: **ESC**

If this doesn't return you to the programming menu, you have not wired a block completely. LOGO! displays the point in the program at which you forgot something (LOGO! only accepts completed programs, which is very much in your interests). Also refer to page 53.

Note

LOGO! has now stored your program permanently, so that it will not be lost in the event of a power failure. The program is stored in LOGO! until you expressly delete it by entering the appropriate command.

---

2. Return to the main menu: **ESC**

Switching LOGO! to RUN

3. Move '>' to 'Start': ▲ or ▼

4. Accept Start: **OK**

LOGO! switches to RUN. In RUN LOGO! displays the following:

LOGO!'s display panel in RUN


```
I:123456
 Mo 09:00

Q:1234 RUN
```

State of the inputs

Current time
(variants with time switch only)

LOGO! is in RUN

State of the outputs

```
I:12345678
9 10 11 12
Mo 09:00
Q:12345678
```

State of the ASi inputs

State of the ASi outputs

State of the ASi bus

```
ASi_Bus
Ia : 1234
Qa : 1234
Bus: On
```
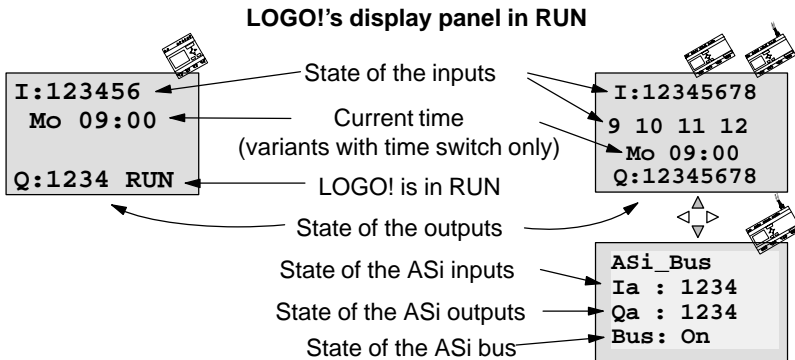
What does "LOGO! is in RUN?" mean?

In RUN, LOGO! executes the program. First of all it reads the states of the inputs, uses the program you have specified to determine the states of the outputs, and switches the relays at the outputs on or off.

LOGO! represents the state of an input or output as follows:

```
I:123456
 Mo 09:00

Q:1234 RUN
```

Input/output has the state '1': inverse

Input/output has the state '0': not inverse

```
I:12345678
9 10 11 12
 Mo 09:00
Q:12345678
```
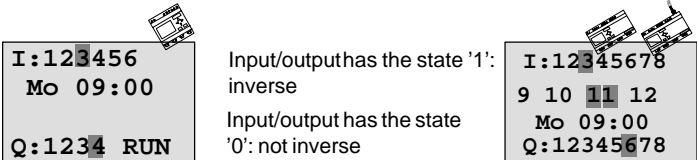
State display

Let's have a look at that in our example:

When switch S1 is closed, voltage is applied to input I1, which has the state '1'.

LOGO! calculates the states of the outputs with the program.

Output Q1 has the state '1' here.

When Q1 has the state '1', LOGO! operates relay Q1 and the consumer at Q1 is supplied with voltage.
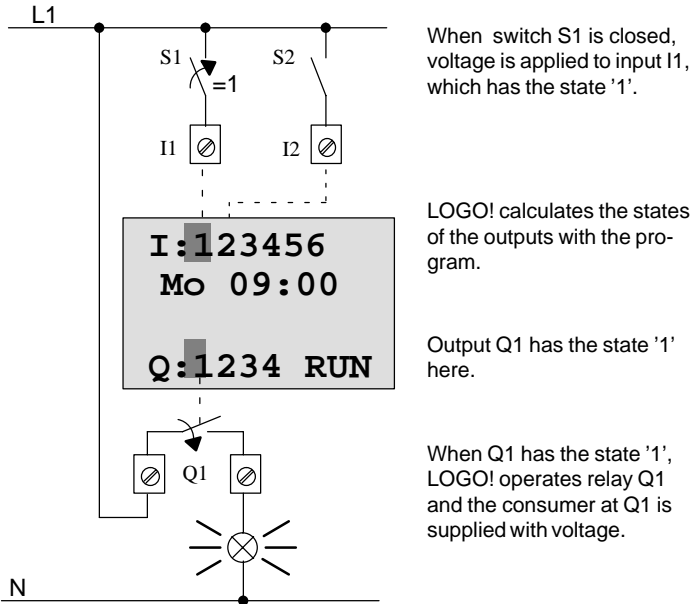
The next step

You have now successfully entered your first circuit.

In the next section, we will show you how to make changes to existing programs and use special functions in them.

3.6.4   Second program

We will use the second program to show you the following:

- How to insert a block in an existing program.
- How to select a block for a special function.
- How to enter parameters.

Changing circuits

In order to produce the second program, we have to modify the first one slightly.

Let's begin by looking at the circuit diagram for the second program:
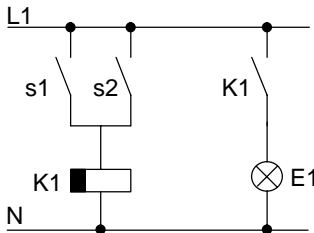
You already know the first part of the circuit. Switches S1 and S2 operate a relay. This relay switches on consumer E1 and switches it off after a delay of 12 minutes.

In LOGO! the program looks like this:

This is the additional new block
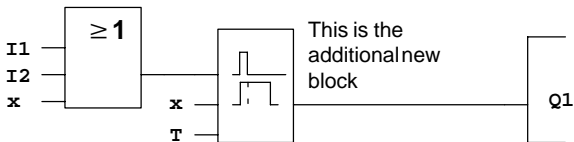
You will recognize the OR block and output relay Q1 from the first program. Only the off-delay is new.

Editing a program

Switch LOGO! to programming mode

To do this, proceed as follows:

1. Switch LOGO! to programming mode:
   (◄, ► and **OK** at the same time).
2. Select "Program.." from the main menu
   (by moving '>' to "Program.." and press **OK**)
3. Select "Edit Prg.." from the programming menu
   (by moving '>' to "Edit Prg.." and press **OK**)

You can now modify the existing program.

Inserting an additional block in a program

Move the cursor to the B of B01 (B01 is the block number
of the OR block).
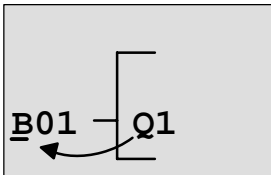
Move the cursor:
Press ◄

At this point we insert the new block. Press **OK**.
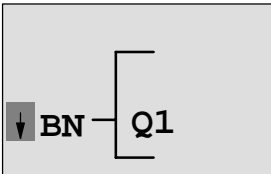
LOGO! displays the BN list.

Select the SF list (▼ key).
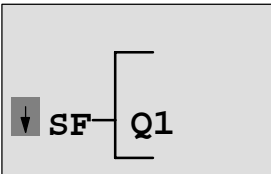
The SF list contains the blocks
for the special functions

Press **OK**.

The block of the first special function appears:

```
Trg ─┤█  ┐
       │  │
  T  ──┤  └─ Q1
```
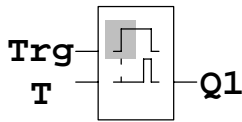
When you select a block for a special or basic function, LOGO! displays the block of the function. The cursor is in the block and appears in the form of a solid block. Use the ▼ or ▲ to select the desired block.

Select the desired block (off-delay, see next diagram) and press **OK**:

```
        ┌──── B02
B01 ────┤   ┐
  R     │   │
  T     └   └─ Q1
```
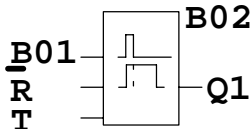
The inserted block receives the block number B02. Block B01, which has been connected up to now to Q1, is automatically connected to the uppermost input of the inserted block. The cursor is positioned at the uppermost input of the inserted block.

The off-delay block has 3 inputs. The uppermost input is the trigger input (Trg). Use this input to start the off-delay. In our example, the off-delay is started by the OR block B01. Reset the time and outputs using the reset input and set the time for the off-delay using the T parameter.

In our example, we don't use the reset input of the off-delay. We wire it with 'x'. You learned how to do this in the first program, but just to remind you, here is the procedure again:

1. Position the cursor under the R: ▲ or ▼
2. Switch to input mode: **OK**
3. Select the Co list: ▲ or ▼
4. Accept the Co list: **OK**
5. Select x: ▲ or ▼
6. Accept x: **OK**

```
        ┌──── B02
B01 ────┤   ┐
  x     │   │
  T     └   └─ Q1
```
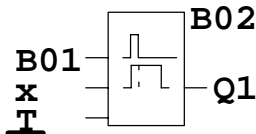
The display should now look like this:

Parameterizing a block

Now enter the time T for the off-delay:

1. If the cursor is not yet under the T, move it there:

$$\blacktriangle \text{ or } \blacktriangledown$$

2. Switch to input mode: **OK**

LOGO! displays the parameter window for parameters:

```
B02:T ◄
T=00.00s+ ◄
```

B02: the parameter of block B02 T: is a time

+ Means: the parameter is displayed in parameteriza-tion mode and can be modi-fied there

Time value

Time unit
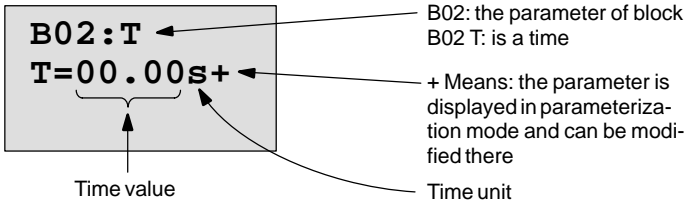
The cursor appears on the first position of the time value.

To change the time value, proceed as follows:

• Use the keys ◄ and ► to move the cursor to different positions.

• Use the keys ▲ and ▼ to change the value.

• When you have entered the time value, press **OK**.

Setting the time

Set the time T = 12:00 minutes:

1. Move the cursor to the first position:
   ◄ or ►

2. Select '1':
   ▲ or ▼

3. Move the cursor to the second position:
   ◄ or ►

4. Select '2':
   ▲ or ▼

5. Move the cursor to the unit:
   ◄ or ►

6. Select the unit m for minutes:
   ▲ or ▼

Displaying/hiding parameters – type of protection

If you don't want the parameter to be displayed in parame-
terization mode, proceed as follows:

7. Move the cursor to the protection mode::
   ◄ or ►

8. Select protection mode '–':
   ▲ or ▼

The following message should now appear on the display:

```
B02:T
T=12:00m+
```

or

```
B02:T
T = 12:00m
```
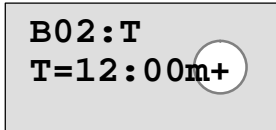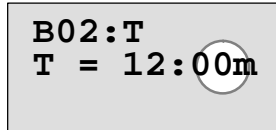
Type of protection+: Time
T can be changed in para-
meterization mode

Type of protection–: Time
T cannot be changed in
parameterization mode

9. Conclude your input:
   OK

Checking the program

This branch of the program for Q1 is now complete.
LOGO! displays the Q1 output. You can have another look
at the program on the display. Use the keys to move
through the program. Use ◄ or ► to move from block to
block and use ▲ and ▼ to move between the inputs on a
block.

Exiting programming mode

Exit program input in the same way you did for the first pro-
gram, but just remind you, here is the procedure again:

1. Return to the programming menu: **ESC**
2. Return to the main menu: **ESC**
3. Move '>' to 'Start':   ▲ or ▼
4. Accept Start: **OK**

LOGO! is now in RUN again:

```
I:123456
 Mo 09:00

Q:1234 RUN
```

3.6.5 Deleting a block

Let's suppose you want to delete block B02 from the fol-
lowing program and connect B01 directly with Q1.
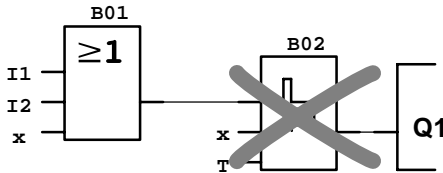
To do this, proceed as follows:

1. Switch LOGO! to programming mode (3-finger grip).
2. Select 'Edit Prg' by pressing **OK.**
3. Position the cursor at the input of Q1, i.e. under B02
   using the key ◄:
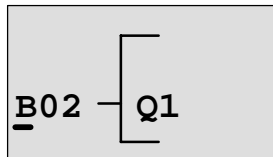
4. Press **OK.**
5. Connect block B01 instead of block B02 directly to out-
   put Q1:
   Select the BN list and press **OK**
   Select B01 and press **OK** .

Result: Block B02 is now deleted, because it is no longer
used anywhere within the entire circuit. Block B01 is now
connected directly to the output instead of B02.

3.6.6 **Deleting a number of interconnected blocks**

Let's suppose you want to delete blocks B01 and B02 from the following program.
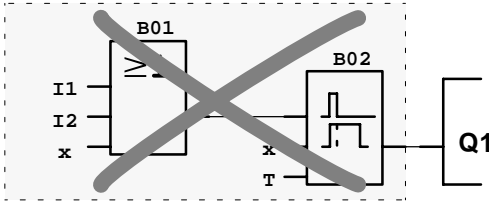
To do this, proceed as follows:

1. Switch LOGO! to programming mode (3-finger grip).
2. Select 'Edit Prg' by pressing **OK.**
3. Position the cursor at the input of Q1, i.e. under B02:
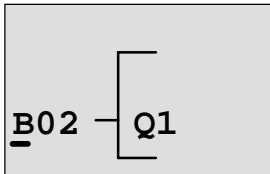
4. Press **OK**
5. Set the connector x instead of block B02 at the Q1 output:
   Select the Co list and then press **OK.**
   Select x and then press **OK**.

Result: Block B02 is now deleted, because it is no longer used anywhere within the entire circuit, and all the blocks that are connected to it are deleted (i.e. block B01 in the example).

3.6.7 **Correcting typing errors**

It is easy to correct typing errors in LOGO!:

- If you have not yet concluded input, you can use **ESC** to go back a step.
- If you have already concluded input, simply start again, as follows:
    1. Move the cursor to the location of the error.
    2. Switch to input mode: **OK**
    3. Enter the correct wiring for the input.

You can only replace one block with another if the new block has exactly the same number of inputs as the old one. However, you can delete the old block and insert a new one. You can insert whichever block you like.

3.6.8 **"?" on the display**

If you have entered a program and want to exit "Edit Prg" with **ESC**, LOGO! checks whether you have wired all the inputs of all the blocks correctly. If you have forgotten an input or parameter, LOGO! displays the first place at which you have forgotten something and marks with a question mark all those inputs and parameters that have not been wired.

You have not yet wired the in-
put here

You have not yet specified a
value for the parameter

B02
B01
R ?   Q1
T ?
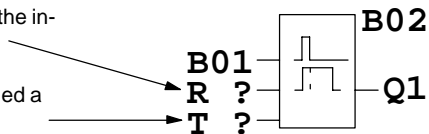
Wire the input and enter a value for the parameter. You can then exit "Edit Prg" by pressing **ESC** .

3.6.9 Deletinga program

To delete a program, proceed as follows:
1. Switch LOGO! to programming mode:
   ◄, ► and **OK** at the same time

```
>Program..
 PC/Card..
 Start
```

2. Move the '>' ▲ or ▼ to 'Program..' using the or key and press **OK**

```
>Edit Prg
 Clear Prg
 Set Clock
```

LOGO! switches to the programming menu.

3. Move the '>' to "Clear Prg":
   ▲ or ▼
4. Accept 'Clear Prg':
   OK

```
 Clear Prg
>No
 Yes
```

To prevent you from inadvertently deleting the program, we have included an additional query.

If you do not want to delete the program, leave the '>' on 'No' and press **OK**.

If you are sure that you want to delete the program stored in LOGO!, proceed as follows:

5. Move the '>' to Yes:
   ▲ or ▼
6. Press **OK**. LOGO! deletes the program.

3.7 Functions

LOGO! offers you a number of elements in programming mode. So that you don't lose track of things, we have divided these elements into lists. These lists are:

- ↓**Co**: List of connectors (**Co**nnector)
  (see Section 3.8)
- ↓**BF**: List of the basic functions AND, OR, ...
  (see Section 3.9)
- ↓**SF**: List of the special functions
  (see Section 3.10)
- ↓**BN**: List of the blocks already configured in the circuit and reusable

Contents of the lists

All the lists display elements available in LOGO!. Normally, these are all the connectors, all the basic functions and all the special functions that the respective LOGO! variant knows. These elements also include all the blocks that you have already generated in LOGO! before you call up the ↓**BN** list.

When LOGO! no longer displays everything

LOGO! no longer displays all elements if:

- No further block must be inserted.

  In this case, there is either no more memory available or the maximum number of possible blocks has been reached (56).
- A special block would use more memory than is still available in LOGO!.
- The resulting number of blocks connected in series would exceed 7 (see Section 3.11).

3.8 Constants and connectors – (Co)

Constants and connectors (Co) describe inputs, outputs, markers and fixed voltage levels (constants).

Inputs

Inputs are identified by the letter **I**. The input numbers (I1, I2, ...) correspond to the numbers of the input connectors on LOGO!.

Inputs Ia1 to Ia4 are also available for communication via the ASi bus in LOGO! variants with an AS interface connection (LOGO!...B11).

Outputs

Outputs are identified by the letter **Q**. The output numbers (Q1, Q2, ...) correspond to the numbers of the output connectors on LOGO!.

Outputs Qa1 to Qa4 are also available for communication via the ASi bus with LOGO! variants with an AS interface connection (LOGO!...B11).

Markers

Markers are identified by the letter **M**. Markers are virtual outputs that have the same value at their output as they have at their input. There are 4 markers (M1 ... M4) available in LOGO!.

By using markers you can exceed the maximum number of consecutive blocks.

Note

The signal applied at the marker's output is always that of the last program cycle. The signal is not changed within a program cycle.

LOGO! Manual
                                                            A5E00046684 01

Levels

Voltage levels are identified by **hi** and **lo**. If a block is supposed to have the state "1" = hi or "0" = continuously, the input is wired with the fixed level or the constant hi or lo value.
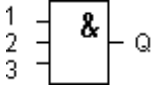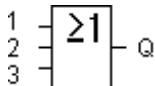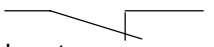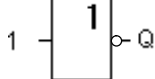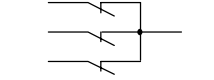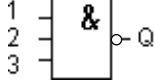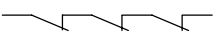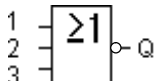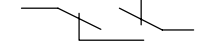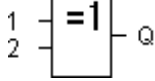
Open connectors

If the connector pin of a block is not to be wired, this is indicated with an **x**.

3.9 Basic functions – BF

Basic functions are simple basic operation links in boolean algebra.

When you enter a circuit, you will find the blocks for basic functions in the BF list. The following basic functions exist:

Circuit diagram representation	Representation in LOGO!	Designation of the basic function
Series connection of normally open contacts	1 2 3 & Q	AND (see page 59)
Parallel connection of normally open contacts	1 2 3 ≥1 Q	OR (see page 59)
Inverter	1 1 Q	NOT (negation, inverter) (see page 60)
Parallel connection of normally	1 2 3 & Q	NAND (and not) (see page 61)
Series connectionof normally closed contacts	1 2 3 ≥1 Q	NOR (or not) (see page 61)
Double change-over	1 2 =1 Q	XOR (exclusive or) (see page 62)

3.9.1 AND

The series connection of a number of normally open contacts is represented in a circuit diagram as follows:
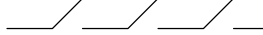
Symbol in LOGO!:
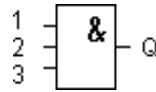
The output of the AND only adopts the state 1 if **all** the inputs have the state 1 (i.e. they are closed).

If an input pin of this block is not wired (x), then the following applies to the input: x = 1.

Logic table for AND

1	2	3	Q
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	1

3.9.2 OR

The parallel connection of a number of normally open contacts is represented in a circuit diagram as follows:
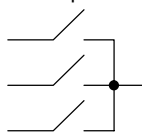
Symbol in LOGO!:
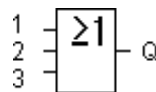
The output of the OR adopts the state 1 if **at least one** input has the state 1 (i.e. it is closed).

If an input pin of this block is not wired (x), then the following applies to the input: x = 0.

Logic table for OR:

1	2	3	Q
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	1

3.9.3 NOT (negation, inverter)

A normally closed contact is represented in a circuit diagram as follows:
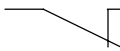
Symbol in LOGO!:
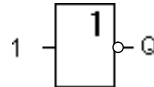
The output adopts the state 1 if the input has the state 0. In other words, NOT inverts the state at the input.

The advantage of the NOT function can be illustrated by the following example: you no longer require normally closed contacts for LOGO! You use a normally open contact and convert it into a normally closed contact by using NOT.

Logic table for NOT

1	Q
0	1
1	0

3.9.4 NAND

The parallel connection of a number of normally closed contacts is represented in a circuit diagram as follows:
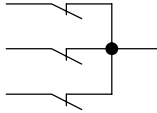
Symbol in LOGO!:
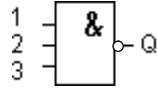
The output of NAND only adopts the state 0 if **all** the inputs have the state 1 (i.e. they are closed).

If an input pin of this block is not wired (x), then the following applies to the input: x = 1.

Logic table for NAND

1	2	3	Q
0	0	0	1
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	0

3.9.5 NOR

The series connection of a number of normally closed contacts is represented in a circuit diagram as follows:
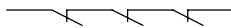
Symbol in LOGO!:
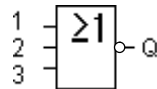
The output of NOR only adopts the state 1 if **all** the inputs have the state 0 (i.e. they are switched off). As soon as any of the inputs is switched on (state 1), the output of NOR is set to 0.

If an input pin of this block is not wired (x), then the following applies to the input: x = 0.

Logic table for NOR

1	2	3	Q
0	0	0	1
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	0

3.9.6 XOR

An XOR in a circuit diagram is a series connection of two change-over contacts:
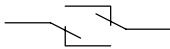
Symbol in LOGO!:
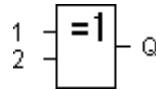
The output of XOR adopts the state 1 if the inputs have **different** states.

If an input pin of this block is not wired (x), then the following applies to the input: x = 0.

Logic table for XOR

1	2	Q
0	0	0
0	1	1
1	0	1
1	1	0

3.10 Special functions – SF

Special functions include time functions, retentivity and various parameterization options to adapt the program to your individual requirements.

3.10.1   Parameter T and time response

Parameter T

In some of the following special functions it is possible to parameterize a T time value. Note the following when you set the time:

Note

Always specify a time for T $\geq$ 0.10 s. For T = 0.05 s and T = 0.00 s the time T is not defined.

Accuracy of T

All electronic components have minute differences. This can result in deviations in the set time (T). In LOGO! the maximum deviation is 1 %.

Example:

In 1 hour (3600 seconds) the deviation is 1% i.e. $\pm$ 36 seconds.

In 1 minute the deviation is therefore only $\pm$ 0.6 seconds.

Accuracy of the time switch

To make sure that this deviation doesn't result in the clock in C variants running inaccurately, the time switch is regularly compared with a high-precision time base and adjusted accordingly.

This results in a maximum time error of $\pm$ 5s per day.
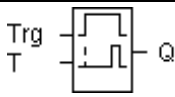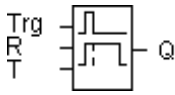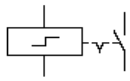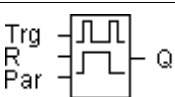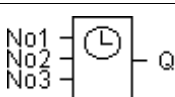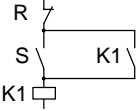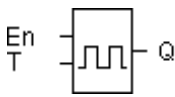
The year 2000 is not a problem for LOGO!, incidentally.
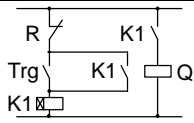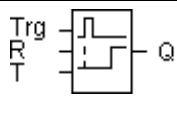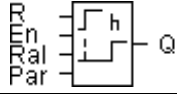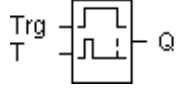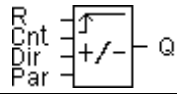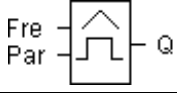
3.10.2 Retentivity

The switching states and counter values can be kept retentively in the special functions. To do this, retentivity must be switched on in the relevant function.

3.10.3 Table of special functions

When you enter a program in LOGO!, you will find the blocks for the special functions in the SF list. In the following table you will also find comparable representations from circuit diagrams as well as an indication as to whether the function in question has parameterizable retentivity.

Circuit diagram representation	Representation in LOGO!	Description of the special function	Re
	Trg T	On delay (see page 67)	
	Trg R T	Off delay (see page 69)	
	Trg R Par	Current impulse relay (see page 71)	Re
	No1 No2 No3	Time switch (see page 72)	
R S K1 K1	R S Par	Latching relay (see page 77)	Re
	En T	Clock pulse generator (see page 79)	

Circuit diagram representation	Representation in LOGO!	Description of the special function	Re
R K1 Trg K1 Q K1	Trg R T Q	Retentive on-delay (see page 81)	
	R En Ral Par Q	Operating hours counter (see page 83)	
	Trg T Q	Interval time-delay relay (see page 86)	
	R Cnt Dir Par +/− Q	Up and down counter (see page 87)	Re
	Fre Par Q	Threshold switch (see page 90)	
	En Inv Par Q	Asynchronous pulse generator (see page 92)	
	No **MM YY** Q	Year time switch (see page 93)	

Reset input R (reset)

In all functions, input R has priority over all other inputs.

Connector X on the inputs of the special functions

If you wire inputs of special functions to the x connector, these inputs will be assigned the value 0 i.e. a low signal is applied to the inputs.

Type of protection:

> The parameter protection setting allows you to specify whether the parameters can be displayed and altered in parameter assignment mode on the LOGO! module. There are two possible settings:
>
> +: The parameter settings are also displayed in parameter assignment mode and can be changed.–: The parameter settings are not displayed in parameter assignment mode and can only be changed in programming mode.

3.10.4 On-delay

Brief description

In the case of the on-delay, the output is not switched over until a definable time period has elapsed.

Symbol in LOGO!	Connection	Description
	Input Trg	Use Trg (trigger) input to start the time for the on-delay.
	Parameter T	T is the time after which the output is switched on (output signal is switched from 0 to 1).
	Output Q	Q switches on after expiry of the defined time T if Trg is still set.

Parameter T

Please note the parameter values for the T parameter in Section 3.10.1.
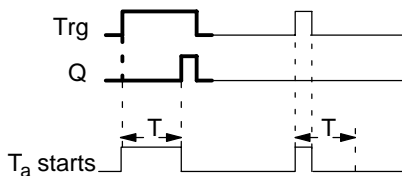
Timing diagram

The bold part of the timing diagram appears in the on-delay symbol.

Description of function

When the state at the Trg input changes from 0 to 1, the time T_a begins to elapse (T_a is the current time in LOGO!).

If the state at the Trg input remains 1 at least for the duration of the parameterized time T, the output is set to 1 after the time T has elapsed (there is a delay between the input being switched on and the output coming on).

If the state at the Trg input switches back to 0 before the time T has elapsed, the time is reset.
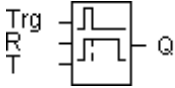
The output is reset to 0 if the state at the Trg input is 0.

In the event of a power failure, the elapsed time is reset.

3.10.5 Off-delay

Brief description

In the case of the off-delay, the output is not reset until a definable time period has elapsed.

Symbol in LOGO!	Connection	Description
Trg R T [symbol] Q	Input Trg	A negative signal edge (change from 1 to 0) at the input Trg (trigger) starts the off-delay timer
	Input R	The input R resets the elapsed time for the off-delay and sets the output to 0.
	Parameter T	T is the period after which the output is switched off (output signal is switched from 1 to 0).
	Output Q	Q switches on when Trg does but when Trg switches off remains on until the delay period T has elapsed.

Parameter T

Please note the parameter values for the T parameter in Section 3.10.1.

Timing diagram

The bold part of the timing diagram appears in the off-delay symbol.
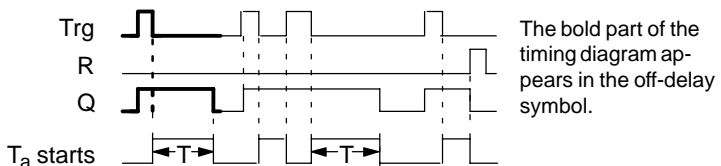
Description of function

> When the state at the input Trg changes to 1, the output Q switches to 1 immediately.

> When the state of Trg changes from 1 to 0, LOGO!'s current time T_a is started and the output remains set. If T_a reaches the values specified via T (T_a=T), the output Q is reset to 0 (off-delay).

> If the Trg input switches on and off again, the time T_a starts again.
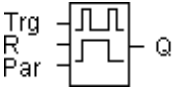
> The input R (Reset) resets the elapsed time T_a and the output before the set time delay T_a has elapsed.

> In the event of a power failure, the elapsed time is reset.

3.10.6 Current impulse relay

Brief description

The output is set and reset by a short pulse at the input.

Symbol in LOGO!	Connection	Description
Trg R Par ⎯ ⊓⊔⊓ ⊓ ⎯ Q	Input Trg	Use the Trg input (trigger) to switch the output Q on and off.
	Input R	The input R resets the current impulse relay and sets the output to 0.
	Parameter Par	This parameter is used to switch retentivity on or off. Rem: off = no retentive feature on = the state can be stored retentively
	Output Q	Q switches on when Trg is sent and remains switched on until T expires.

Timing diagram

The bold part of the timing diagram appears in the current impulse relay symbol.
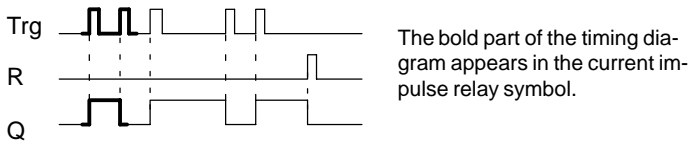
Description of function

Every time the state at the input Trg changes from 0 to 1, the output Q changes its state, i.e. the output is switched on or off.
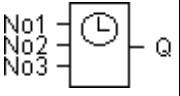
Use the R input to reset the current impulse relay to its initial state (i.e. the output is set to 0).

After power failure the current impulse relay is reset and the Q output changes to 0 if you haven't switched on the retentive feature.

3.10.7 Time switch

Brief description

Up to three time windows in which the output is active can be defined.

Symbol in LOGO!	Connection	Description
No1 No2 No3 — Q	Parameters no 1, no 2 No 3	Use the No parameters to set the on and off times for each of the cams **No** of the time switch. You specify the times in terms of day and time.
	Output Q	Q switches on when a para-meterized cam is switched on.

A time switch is only available in LOGO! variants that have the letter **C** for clock in their name (e. g. LOGO! 230R**C**).

Timing diagrams (3 examples)
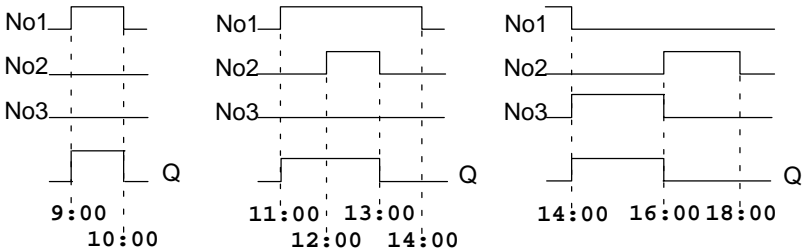
Description of function

Each time switch has three setting cams, each of which can be used to configure a time window.

You use the cams to set the on and off times for the time windows. At the on times, the time switch switches the output on if it is not already on. At the off times, the time switch switches the output off if it is not already off.

If you set an on time on one cam that is the same as the off time on another cam on the time switch, then the on and off times conflict. In such cases, cam 3 has priority over cam 2 and cam 2 has priority over cam 1.

Parameter window

The parameter window for cam no. 1, for example, is as follows:

```
B01:No1
Day=Mo    +
On =06:00
Off=19:00
```

Block B01   Cam No 1

Day of the week (Mo for Monday)

See the section "Displaying/ hiding parameters – type of protection" on page 49

On time (6.00 hours)

Off time (19.00 hours)
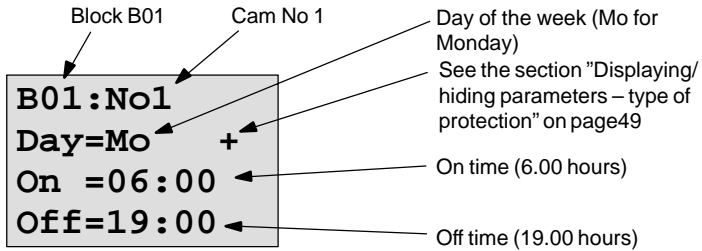
Day of the week

The following options are available to you for setting the days of the week:

- Su   Sunday
- Mo   Monday
- Tu   Tuesday
- We   Wednesday
- Th   Thursday
- Fr   Friday
- Sa   Saturday
- Mo..Fr   From Monday to Friday
- Mo..Sa   From Monday to Saturday
- Mo..Su   From Monday to Sunday (i.e. every day)
- Sa..Su   Saturday and Sunday

Switching times

> Any time between 00:00 and 23:59 hours
> —:— means there is no on time
>
> Any time between 00:00 and 23:59 hours
> —:— means there is no off time

Setting the time switch

> To enter the switching times, proceed as follows:
>
> 1. Position the cursor on one of the time switch's No pa-
>   rameters (e. g. No1).
> 2. Press **OK**. LOGO! opens the parameter window for the
>   cam. The cursor is positioned on the day of the week.
> 3. Use the ▲ and ▼ keys to select one or more days of the
>   week.
> 4. Use the ► key to move the cursor to the first position for
>   the on time.
> 5. Set the on time.
>   Use the ▲ and ▼ keys to change the value. To move the
>   cursor from one position to another, use the ◄ and ►
>   keys. You can only select the value —:—
>   (—:— means no switching operation).
> 6. Use the ► key to move the cursor to the first position for
>   the off time.
> 7. Set the off time (same procedure as for step 5).
> 8. Conclude your input by pressing **OK**.
>
>   The cursor is positioned at parameter No 2 (cam 2).
>   You can now parameterize another cam (as described
>   under points 2. to 8.).

Note

> Please refer to the technical specifications in Sec-
> tion 3.10.1 for information regarding the accuracy of the
> time switch.

Clock: examples

You can use the time switch to combine on and off times however you like.

Here are some examples:

Example 1

The time switch's output is to be switched on every day (i.e. from Monday to Sunday) from 08:00 to 13:00.

```
B01:No1
Day= Mo..Su                              On
On =08:00
Off=13:00        8:00    13:00           Off
```
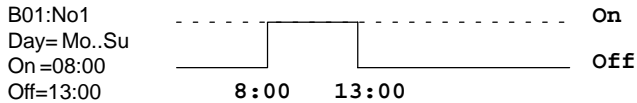
Example 2

The time switch's output is to be switched on every day from 08:00 to 13:00 and from 15:00 to 18:30. You need 2 cams for this:

```
B01:No1      B01:No2                          On
Day= Mo..Su  Day= Mo..Su
On =08:00    On =15:00      8:00     15:00     Off
Off=13:00    Off=18:30         13:00     18:30
```
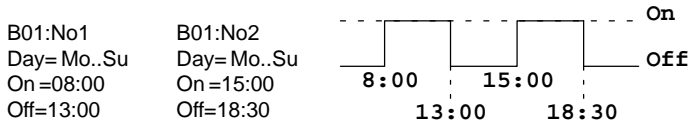
Example 3

The time switch's output is to be switched on every day from Monday to Saturday from 08:00 to 13:00 and from 15:00 to 18:30. It is also to be switched on between the hours of 11:00 and 15:00 on Sundays. You need 3 cams for this:

```
              Mo...Sa          B01:No1      B01:No2
On                             Day= Mo..Sa  Day= Mo..Sa
Off                            On =08:00    On =15:00
     8:00    15:00             Off=13:00    Off=18:30
        13:00    18:30

              Su               B01:No3
On                             Day= Su
Off                            On =11:00
     11:00   15:00             Off=15:00
```
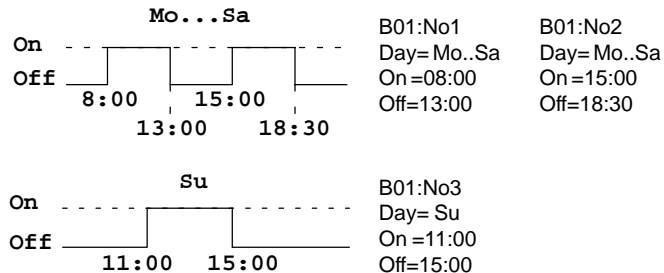
Example 4

The time switch's output is to be switched on on Monday at
22:00 hours and of on Tuesday at 6:00 hours.
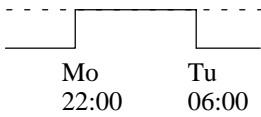
		B01:No1	B01:No2
		Day= Mo	Day= Tu
Mo	Tu	On =22:00	On =—:—
22:00	06:00	Off=—:—	Off=06:00

3.10.8 Latching relay

Brief description

The output Q is set via the input S. The output is reset via the input R.

Symbol in LOGO!	Connection	Description
	Input R	The input R resets the out-put Q to 0. If S and R are both 1 at the same time, then the output is reset
	Input S	Input S sets the output Q to 1.
	Parameter Par	This parameter is used to switch retentivity on and off. Rem: off = no retentive feature on = the state can be stored retentively
	Output Q	Q switches on when S does and remains on until the in-put R is set.

Timing diagram
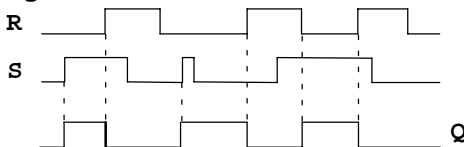
Switching behavior

A latching relay is a simple binary memory cell. The signal at the output depends on the states of the inputs and the previous state at the output. The following table illustrates the logic once more:
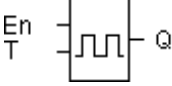
S_n	R_n	Q	Remarks
0	0	x	State remains the same
0	1	0	Reset
1	0	1	Set
1	1	0	Reset (resetting has priority over setting)

After power failure the signal that was valid before the power failure is set at the output if the retentive feature was switched on.

3.10.9 Symmetrical clock pulse generator

Brief description

A clock pulse with a definable period length is output at the output.

Symbol in LOGO!	Connection	Description
En T ‒⊓⊔‒ Q	Input En	The En input switches the clock pulse generator on and off.
	Parameter T	T is the time for which the output is switched on or off.
	Output Q	Q switches on and off cyclically in time with the pulse period T.

Parameter T

Please refer to the note in Section 3.10.1 when specifying the values.

Timing diagram

En ⎺⎺⎺⎺⎺⎺⎺⎺⎺⎺⎺⎺⎺⎺⎺

Q ⎯⎯⎯⎺⎺⎺⎯⎯⎯⎺⎺⎺⎯⎯⎯⎺⎺
⟵T⟶ ⟵T⟶ ⟵T⟶ ⟵T⟶
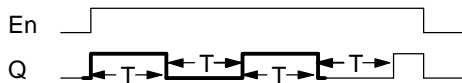
The bold part of the timing diagram appears in the symmetrical clock pulse generator symbol.

Description of function

The parameter T specifies the length of the on and off periods. Use the En (enable) input to switch the clock pulse generator on. The pulse generator sets the output to 1 for the time T, then to 0 for the time T and so on until the En input is a 0.
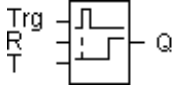
Note on the relay ouputs:

Relay outputs that switch under load get worn a little with each switching operation. To find out how many switching operations a LOGO! output can execute, refer to the Chapter entitled Technical Specifications (see Chapter A).

3.10.10 Retentive on-delay

Brief description

Following an input pulse, a definable timed period starts on expiry of which the output is set.

Symbol in LOGO!	Connection	Description
Trg R T	Input Trg	Use the Trg (trigger) input to start the time for the on-delay.
	Input R	Input R resets the time for the on-delay and sets the output to 0.
	Parameter T	T is the time after which the output is switched on (output state is switched from 0 to 1).
	Output Q	Q is switched on when the delay period T has elapsed.

Parameter T

Please refer to the note in Section 3.10.1 when specifying the values.
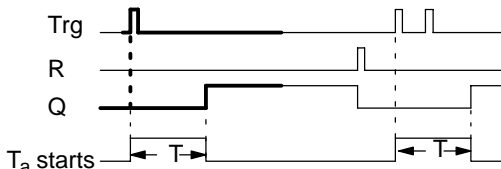
Timing diagram

The bold part of the timing diagram appears in the retentive on-delay symbol.

Description of function

When the state at the Trg input changes from 0 to 1, the current time T_a starts. When T_a reaches the time T, output Q is set to 1. Another switching operation at the Trg input has no effect on T_a.
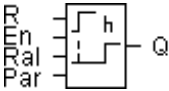
The output and the time T_a are not reset to 0, until the state of the R input changes to 1 again.

In the event of a power failure, the elapsed time is reset.

3.10.11 Operating hours counter

Brief description

When the input is set, a definable timed period starts running. The output is set when the defined period has elapsed.

Symbol in LOGO!	Connection	Description
R ⎯⎤ h ⎤ En ⎯⎤ ⎤⎯ Q Ral ⎯⎤ ⎤ Par ⎯⎦ ⎦	Input R	**R = 0:** Hours can be counted if Ral is not 1 **R = 1:** The counter is stopped The input R resets the output. The remaining period of the service interval MN is set to MN = MI. The previously elapsed time remains stored.
	Input En	En is the monitoring input. LOGO! measures the length of time in which this input is set.
	Input Ral	**Ral = 0:** Hours can be counted if R is not 1 **Ral = 1:** The counter is stopped The input Ral (Reset all) resets the counter and the output, i.e. • Output Q = 0, • Recorded operating time OT = 0 • Remaining period of service interval MN = MI.

Symbol in LOGO!	Connection	Description
	Parameter Par: MI	**MI:** specified service interval in hours. MI can be any number of hours in the range 0 to 9999.
	Output Q	If the remaining time MN = 0 (see timing diagram), the output is set.

MI =   parameterized count value
MN=   remaining time
OT=   total time elapsed since the last 1 signal at the
        Ral input

Timing diagram

R
En
Ral
                                                    Q
MN=MI   Par:
                                                    MI=5h
MN=0

                                        1h   x   x–R=1h
OT
                                              R

█████   The counter stops counting as long as R or Ral is set
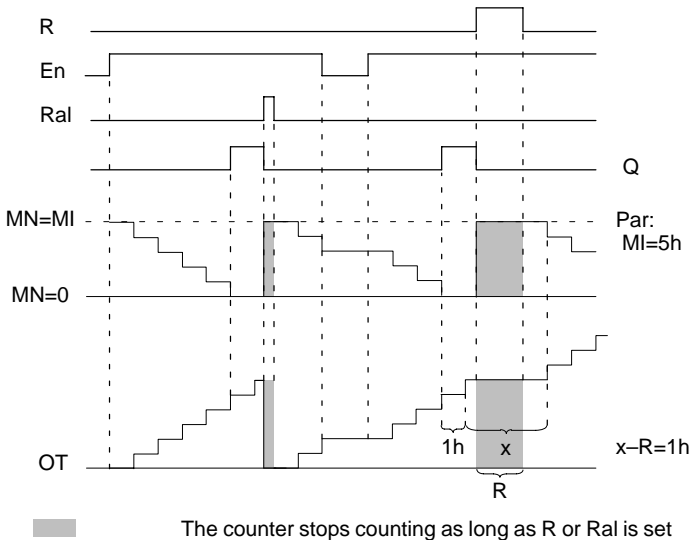
MI = Defined time period
MN = Remaining period
OT = Total elapsed time since last 1 signal at input Ral

Description of function

> The operating hours counter monitors the input En. As long as the signal 1 is present at that input, LOGO! monitors the elapsed time and the remaining period MN. LOGO! displays those times in parameterization mode. When the remaining period MN reaches 0, the output Q is set to 1.

> The reset input R resets the output Q and the timer for the remaining period to the specified period MI. The internal counter OT continues to run.

> The Ral reset input resets the output Q and the counter for the remaining time to the specified value MI. The internal counter OT is reset to 0.

> You can look at the current values for MN and OT during program execution in parameterization mode.

Limit value for OT

> When you reset the hours counter with the signal R, the accumulated time remains stored in OT. The maximum limit for the counter OT is 99999 hours.

> If the hours counter reaches that figure, no more hours are recorded.

Par parameter setting

```
B03:Par


MI = 0000h+
```

— Monitoring interval in hours

— Type of protection
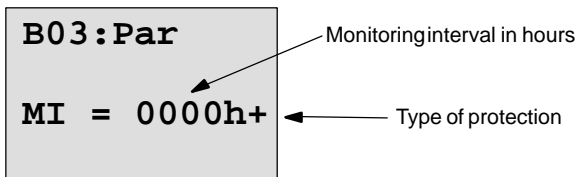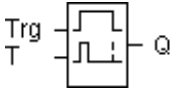
> **MI** is the parameterizable time interval. It can be anything between 0 and 9999.

3.10.12 Interval time-delay relay – pulse output

Brief description

An input signal produces a signal of definable duration at the output.

Symbol in LOGO!	Connection	Description
Trg T — Q	Input Trg	The input Trg (trigger) starts the time for the interval time-delay relay.
	Parameter T	T is the period after which the output is switched off (output signal is switched from 1 to 0).
	Output Q	Q switches on when Trg does but when Trg switches off remains on until the delay period T has elapsed.

Parameter T

Please refer to the note on the T parameter in Section 3.10.1.

Timing diagram

Trg

Q

T_a starts → T ← → ← T has not expired completely

The bold part of the timing diagram appears in the interval time-delay relay symbol.
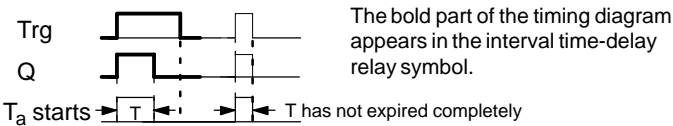
Description of function
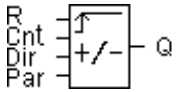
When the input Trg switches to the state 1, the output Q immediately switches to the state 1. At the same time, the elapsed time T_a starts running; the output remains set. If T_a reaches the time specified for T (T_a=T), the output Q is reset to 0 (pulse output).

If the state at the input Trg switches back from 1 to 0 before the specified time has elapsed, the output also immediately switches back from 1 to 0.

3.10.13 Up/down counter

Brief description

On receipt of an input pulse, an internal counter starts counting either up or down, depending on the parameter- ization. When the definable count value is reached, the out- put is set. The direction of the count is set by a separate input.

Symbol in LOGO!	Connection	Description
R Cnt Dir +/− Q Par	Input R	The R input can be used to reset the internal count value and the output to zero.
	Input Cnt	The counter counts the changes from state 0 to state 1 at the Cnt input. Changes from state 1 to state 0 are not counted. Maximum count frequency at the input connectors: 5 Hz
	Input Dir	The Dir input allows you to specify the counting direc- tion as follows: Dir = 0: count up Dir = 1: count down
	Parameter Par	Par is the count threshold value; when the internal counter reaches that value, the output is set.
	Output Q	Q switches on when the count value is reached.

Timing diagram
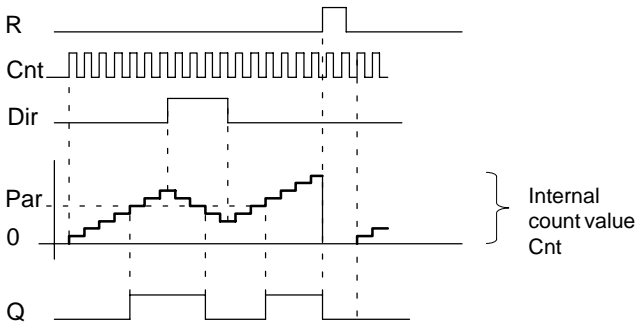
Description of function

At each positive signal edge at the input, Cnt either in-creases the internal counter value by one (Dir = 0) or de-creases it by one (Dir = 1).

If the internal counter value is greater than or equal to the value specified for Par, the output Q is set to 1.

The Reset input R can be used to reset the internal counter value and the output to '000000'. As long as R=1, the out-put remains set to 0 and the pulses at the input Cnt are not counted.

Par parameter setting

Count value

See the parameter for display-ing/hiding – type of protection on page 49

Retentivity

If the internal counter value is greater than or equal to Par, output Q is set. The counter remains the same in the case of under or overflow.

Par can be anything between 0 and 999999.

Rem: This parameter switches the retentive feature of the internal Cnt counter on an off.
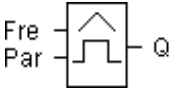
off = No retentive feature
on = The Cnt counter can be stored retentively

If the retentive feature is switched on, when a power failure occurs the counter status remains the same and this value is used once power is restored.

3.10.14 Threshold switch for frequencies

Brief description

The output is switched on or off depending on whether the frequency at the input is between two definable limits.

Symbol in LOGO!	Connection	Description
Fre Par — Q	Input Fre	At the input Fre you apply the input that supplies the pulses to be counted. Use • Inputs I5/I6 or I11/I12 (LOGO!...L) for rapid counts (12/24 V inputs): max. 1 kHz. • Any other input or circuit component for lower frequencies.
	Parameter Par: SW↑, SW↓ G_T	SW↑: On threshold SW↓: Off threshold G_T: time period over which the signal pulses applied are counted.
	Output Q	Q switches on or off depending on SW↑ and SW↓.

Timing diagram

Q

G_T   SW↑ = 9

fre   SW↓ = 5

fa = 6   fa = 10   fa = 8   fa = 5
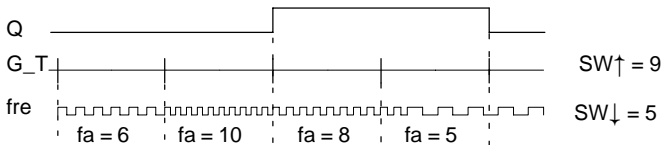
Description of function

> The threshold switch measures the signals at the input Fre. The pulses received are recorded over a definable period G_T. If the frequency of the pulses received in the period G_T is **greater** than the on and the off thresholds, the output Q switches on.

> Q switches off again if the measured pulse frequency **reaches or falls below the off threshold**.

Par parameter setting

```
B03:Par
SW↑=0050+
SW↓=0048
G_T=01.00s
```

- On threshold
- Type of protection
- Off threshold
- Time interval for pulses
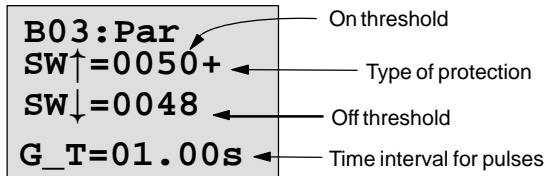
SW↑ is the on threshold. It can be any frequency in the range 0000 to 9999.

SW↓ is the off threshold. It can be any frequency in the range 0000 to 9999.

G_T is the time interval in which the pulses applied to Fre are measured. G_T can be a period between 00.05s and 99.95s.

Note
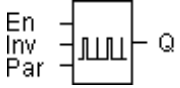
> If you specify the time G_T a 1 s, LOGO! returns the current frequency (in Hz) in the fa parameter.

> fa is always the sum of the pulses measured per time unit G_T.

3.10.15 Asynchronous pulse generator

Brief description

The pulse form of the output can be modified by means of the definable pulse duration/interval ratio.

Symbol in LOGO!	Connection	Description
En Inv Par — ⎍⎍⎍⎍ — Q	Input En	The En input switches the asynchronous pulse generator on and off.
	INV input	The input INV is used to invert the output signal of the asynchronous clock pulse generator when active.
	Parameter Par	The pulse duration TH and pulse interval TL can be set.
	Output Q	Q switches on and off cyclically on the basis of the pulse timing parameters TH and TL.

Timing diagram
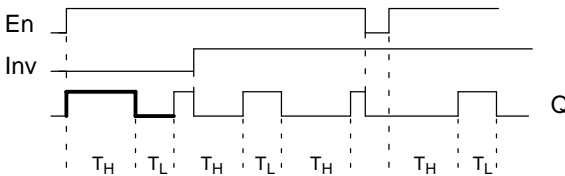
Description of function

The pulse duration and pulse interval can be set using the TH (Time High) and TL (Time Low) parameters. Both parameters have the same time basis, they cannot be set individually to different bases.

The input INV allows the output to be inverted. The input INV only negates the output if the block is activated by EN.

3.10.16 Twelve-month time switch

Brief description

The output is controlled by a definable on/off date.

Symbol in LOGO!	Connection	Description
No — [MM YY] — Q	Input No	Use the No parameter to set the on and off times for the cam of the twelve-month time switch.
	Output Q	Q switches on when the parameterizable cam is switched on.

Timing diagram

```
              feb.        mar.        apr.
MM.DD+     - - - - _____ - - - -      On
On=02.20   _____|                |_____     Off
Off=04.03

            20 February      3 April
```
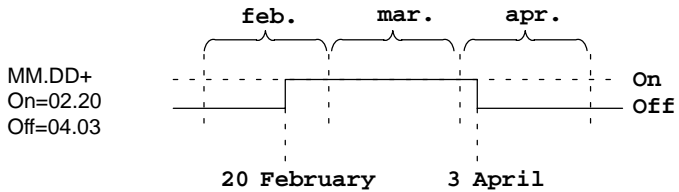
Description of function

At the on time, the year time switch switches the output on and at the off time, it switches the output off. The off date indicates the date on which the output is reset to 0. The first value indicates the month, the second value indicates the day.

The LOGO! module is not susceptible to year-2000 problems.

Parameterization example

The output of a LOGO! has to be switched on every year on 1 March and switched off on 4 April, switched on again on 7 July and switched off on 19 November. For this you require 2 twelve-month time switches each of which is configured for one of the on periods. The outputs are then linked by an OR block.

```
B01:No      B02:No          ┌─────────┐    ┌─────────┐ ── On
   MM.DD        MM.DD        │   B01   │    │   B02   │
On=03.01    On=07.07    ────┘         └────┘         └──── Off
Off=04.04   Off=11.19   1 March     7 July
                              4 April    19 November
```
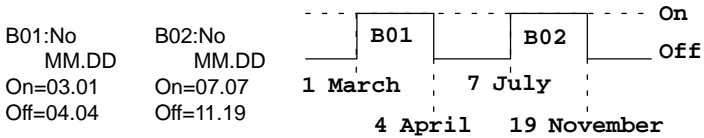
Clock backup

The internal clock of a LOGO! module continues to run if there is a power failure, i.e. it has a power buffer. The duration of the reserve power is affected by the ambient temperature. At an ambient temperature of 25°C the typical power buffer is 80 hours.

3.11 Memory required and size of a circuit

A program (control program in LOGO!, circuit diagram) has the following limitations:
- The number of blocks connected in series (nesting depth)
- The memory space (use of memory by the blocks)

Memory area

You can use a certain number of blocks for your program in LOGO!. In addition, some blocks require extra memory for their special functions.

The memory required for special functions is divided up into 4 memory areas.
- Par: The area in which LOGO! stores the desired values (e.g. the limit values of a counter).
- RAM: The area in which LOGO! stores the current actual values (e.g. the counter status).
- Timer: The area LOGO! uses for time functions such as the off-delay.
- REM: The area in which LOGO! stores actual values that have to be retained (e.g. the count value of an operating hours counter). In blocks with selective use of the retentive feature, this memory area is only used if the retentive feature is switched on.

Resources available in LOGO!

The maximum amount of resources that can be occupied by a program in LOGO! is as follows:

Blocks	Par	RAM	Timer	REM	Bit
56	48	27	16	15	4

LOGO! monitors memory utilization, and offers in the function lists only those functions for which there is still actually enough memory space available.

Use of memory

The table gives you an overview of the special memory requirements of the special functions:

Function block	Par	RAM	Timer	REM
On-delay	1	1	1	0
Off-delay	2	1	1	0
Current impulse relay*	0	(1)	0	(1)
Time switch	6	2	0	0
Latching relay*	0	(1)	0	(1)
Clock pulse generator	1	1	1	0
Retentive on-delay	2	1	1	0
Operating hours counter	2	0	0	4
Interval time-delay relay	1	1	1	0
Up and down counter	2	(2)	0	(2)
Threshold switch	3	1	1	0
Asynchronous pulse genera-tor	3	1	1	0
Year time switch	2	0	0	0

* Depending on the parameterization of the function with or without retentivity, each function occupies the following amount of memory:

- Retentivity switched off: RAM memory
- Retentivity switched on: REM memory

The maximum number of functions that can be used

Based on the memory requirements of the individual special functions, you can work out the maximum number of special functions you can use.

Example: The operating hours counter requires 2 memory areas for desired value storage (Par) and 4 memory areas for the actual values that have to be retained (REM). There are 15 REM memory areas and 48 Par memory areas In LOGO!.

The special function of the operating hours counter can therefore only be used a maximum of 3 times, leaving just 3 REM memory areas left. Although there are still 42 Par memory areas free, you are one REM memory area short to run an additional operating hours counter.

Calculation: Free memory areas divided by the required number of memory areas. Carry out this calculation for each memory area required (Par, RAM, timer, REM). The lowest value shows you the maximum number of functions you can use.

Nesting depth

In LOGO! there is a maximum nesting depth that can be implemented in your program. The maximum nesting depth in a program path is 9 blocks.

A program path consists of a series of function blocks starting and ending with a terminal block. The number of blocks in a program path describes the nesting depth.

Inputs and levels (I, Ia, hi, lo), as well as outputs and markers (Q, Qa, M) are terminal blocks . The terminal blocks are not represented by a block symbol in LOGO!.

Calculating the nesting depth

If you deduct the two terminal blocks from the maximum nesting depth of 9 blocks, you are left with 7 function blocks, which you can connect in series in a circuit.
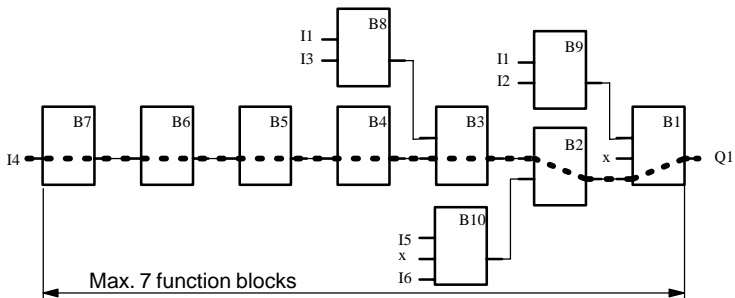
Implementing longer program paths

Longer program paths can be implemented using markers. This is done by placing a marker block at the end of a path. The remaining function blocks are connected to the output of the marker. LOGO! interprets the marker as the output of the first program path (markers are terminal blocks) and also as the input of the next program path. They represent two individual paths to LOGO!.

If another marker block is not available, you can use an output block instead.

4 Parameterizing LOGO!

By parameterization we mean setting the parameters of the blocks. You can set delay times for time functions, switching times for time switches, the threshold value of a counter, the monitoring interval of an operating hours counter and the on and off thresholds of the threshold switch.

You can set the parameters:

- In programming mode
- In parameterization mode

In parameterization mode the programmer sets a value for a parameter.

We introduced parameterization mode so that parameters can be changed without having to change the program. In this way, a user can change the times without having to change into programming mode. The advantage of this is that the program (and thus the circuit) is protected but can still be modified by the user of the circuit to suit requirements.

Note

LOGO! continues to execute the program in parameterization mode.

4.1 Switching to parameterization mode

To switch to parameterization mode, press **ESC** and **OK** at the same time:

```
I:123456
 Mo 09:00

Q:1234 RUN
```
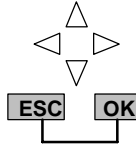
LOGO! switches to parameterization mode and displays the parameterization menu:

```
>Set Clock
 Set Param
```

The 'Set Clock' menu item is executed only if your version of LOGO! has a clock/time switch (those versions of LOGO! that have a clock have the letter C in their name e. g. LOGO 230RC). Set Clock allows you to set LOGO!'s time switch.

4.1.1 Parameters

The following can be parameters:

- The delay times of a time relay.
- The switching times (cams) of a time switch.
- The threshold value of a counter.
- The monitoring time of an operating hours counter.
- The switching thresholds of a threshold switch.

Each parameter is identified by the block number and the parameter abbreviation. Examples:

B01:T

Block number   Parameterabbreviation
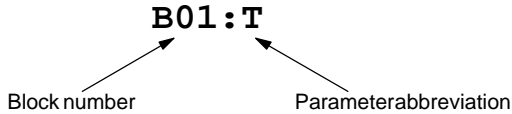
- T: ...is a time that can be set.
- No1: ...is the first cam of a time switch (clock).
- Par: ...refers to several counters that can be monitored.

4.1.2 Selecting a parameter

To select a parameter, proceed as follows:

1. Select the 'Set Param' option from the parameterization menu.

```
  Set Clock
 >Set Param
```
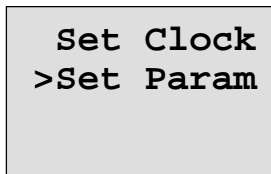
2. Press **OK**

LOGO! displays the first parameter:

```
 B01:T
 T = 12:00m


 Ta= 00:00m
```

Parameter

Value set for the parameter

Current time in LOGO!
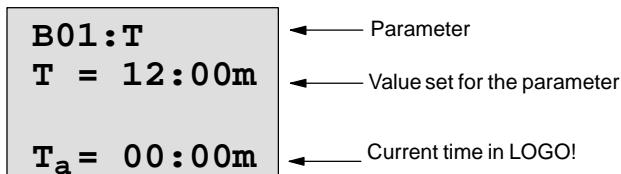
If no parameter can be set, LOGO! displays the following:

```
No Param
Press ESC
```
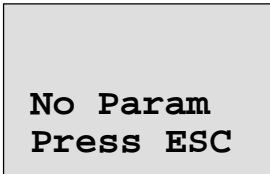
No parameter can be changed:
ESC returns you to the parameterization menu

3. Select the desired parameter:
   ▲ or ▼

LOGO! displays a parameter in a separate window.

4. To change a parameter, select it and press **OK**.

4.1.3 **Changing a parameter**

To change a parameter, you first have to select it (see "Selecting a parameter).

You change the value of the parameter in the same way as you entered it in programming mode:

1. Move the cursor to the point at which you want to make the change:
   ◀ or ▶

2. Change the value:
   ▲ or ▼

3. Accept the value:
   OK

```
B01:T
T = 01:00m


Ta= 00:00m
```

— Move: ◀ or ▶

Finished: **OK**
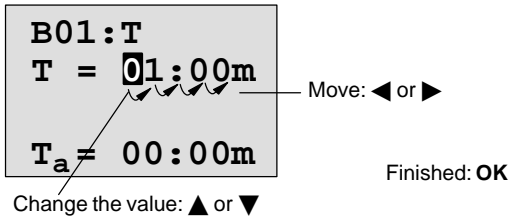
Change the value: ▲ or ▼

You cannot change the unit of the delay time for the parameter T in parameterization mode. This is only possible in programming mode.

Current value of a time T

If you view a time T in parameterization mode, it looks like this:

```
B01:T
T = 12:00m     ◄──── Time T set



Ta= 00:00m     ◄──── Current time Ta
```
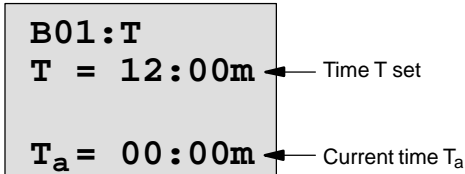
You can change the set time T (see "Changing a parameter").

Current value of the time switch

If you view a cam of a time switch in parameterization mode, it looks like this, for example:

```
B02:No1 ▉1
Day = Su
On =09:00
Off=10:00
```

The circuit state of the time switch is displayed:

0 The time switch is off (state '0' at the output)

1 The time switch is on (state '1' at the output)
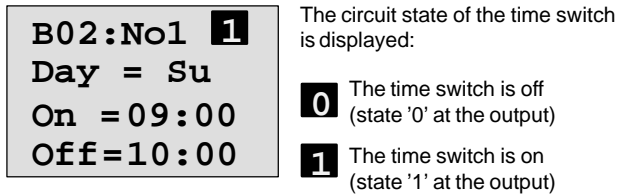
LOGO! displays the circuit state of the time switch rather than the circuit state of a cam. The circuit state of the time switch depends on all three cams (No1, No 2 and No 3).

Current value of a counter

If you view the parameter of a counter in parameterization mode, it looks like this:

```
B03:Par
Lim=000300◄─── Switching threshold


Cnt=000028◄─── Current count value
```
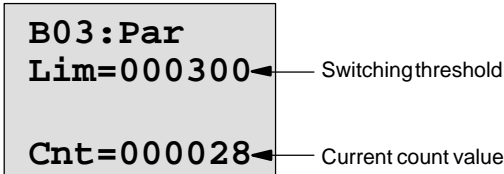
Current value of an operating hours counter

If you view the parameters of an operating hours counter in parameterization mode, it looks like this:

```
B05:Par
MI = 0050h◄─── Monitoring time
MN = 0017h◄─── Remaining time
OT =00083h◄─── Operating hours elapsed
```
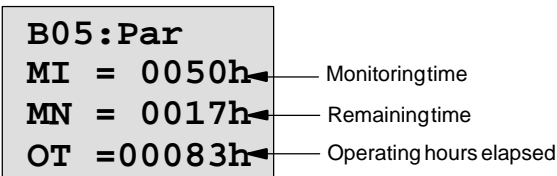
Current value of a threshold switch

If you view the parameter of a threshold switch in parameterization mode, it looks like this:

```
B06:Par
SW↑=0050    ◄─── On threshold
SW↓=0048    ◄─── Off threshold
fa =0012    ◄─── Measured value
```
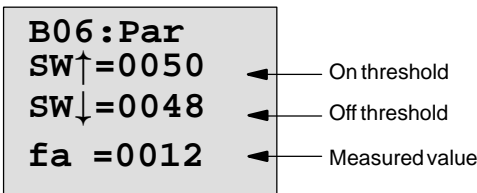
4.2 Setting the time (LOGO! ... C)

You can set the time:
- In parameterization mode
- In programming mode

Setting the time in parameterization mode:

1. Switch to parameterization mode:
   ESC and **OK** at the same time
2. Select 'Set Clock' and press **OK**.

```
Set Clock
_Mo 14:26
 MM.DD.YY
 06.14.99
```

The cursor is positioned before the day of the week.

3. Select the day of the week:
   ▲ or ▼
4. Move the cursor to the next position:
   ◄ or ►
5. Change the value at this position:
   ▲ or ▼
6. Set the clock to the correct time. Repeat steps 4 and 5.
7. Conclude your input:
   OK

Setting the time in programming mode:

1. Switch to programming mode:
   ◄, ► and **OK** at the same time
2. Select 'Program' and press **OK**
3. Select (using ▼ or ▲) 'Set Clock' and press **OK**

Now you can set the day of the week and the time, as described above (as of step 3.).

Switching between summer and winter time:

LOGO! must be in RUN if you want to switch over the time.

1. If necessary, exit programming or parameterization mode and switch LOGO! to RUN.

```
I:12345678
9 10 11 12
  Mo 09:17  ◄──────────  Time displayed
Q:12345678
```
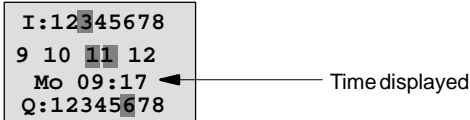
2. Press **OK** and ▲

The current time is put forward by one hour.

```
I:12345678
9 10 11 12
  Mo 10:17  ◄──────────  Time displayed after
Q:12345678                change
```

Change the time in the opposite direction in the same way:

3. Press **OK** and ▼

The current time is put back by one hour.

```
I:12345678
9 10 11 12
  Mo 09:17  ◄──────────  Time displayed after
Q:12345678                change
```

5 LOGO!'s program modules

You can only keep one program stored in LOGO!. If you want to change the program or write another program without deleting the first one, you must archive it somewhere. You can use program modules/cards to do this.

You can copy the program stored in LOGO! to a program module/card. You can insert the program module/card in a different LOGO! and in this way copy the program to the other LOGO!. You can use the program module/card to do the following:

- Archive programs
- Duplicate programs
- Send programs by post
- Write and test programs in the office and then transfer them to a different LOGO! in the cabinet.

LOGO! is supplied with a cover. You receive the program module/card separately.

Note

You do **not** require a module for permanently storing the program in your LOGO!.

The LOGO! program is already stored permanently when programming mode is finished.

Below you will find the two modules that you can get for LOGO!. Both of them can accommodate the entire program memory of a LOGO!.

Module	Order number
Yellow module: for copying	6ED1 056–1BA00–0AA0
Red module: with know-how protection and copy protection feature	6ED1 056–4BA00–0AA0

5.1 Overview of the modules

In this section we will describe the features of the two modules.

Yellow program module

You can read and write programs with this module.

Red program module

You can only write programs with this module. Once a program has been stored, it cannot be looked at, copied or changed (i.e. your data is protected).

Programs saved like this can only be run if the module remains inserted in LOGO! while the system is in operation.

⚠ **Warning**

Make sure that you don't save your program on a module with program protection if you intend to edit it further.

You can start up a program on a module with know-how protection but not read it for the purpose of editing.

Upward compatibility

The modules are only upwardly compatible. This means:

- A module written in a standard variant can be read in all the other variants.
- A module written in a LOGO! ...L variant can be read in all the other LOGO! ...L variants; but not into a standard variant.
- A module written in a LOGO! ...LB11 variant can be read in all the other LOGO! ...LB11 variants; but not in a standard variant or in a LOGO! ...L variant.

5.2 Removing and inserting modules

When you remove a red program module (know-how protection and copy protection), always remember that the program stored on the module can only run if the module is inserted and remains inserted for the entire time. If the module is removed, LOGO! reports 'no program'.

Always heed the following warning:

⚠ **Warning**

Do not put your finger or an object made of metal or any other conductive material in the open shaft of the program module/card.

The socket for the program module/card may be live if L1 and N have been incorrectly wired.

The program module/card should only be changed by a trained technician.

Removing the module

Remove the program module/card as follows:
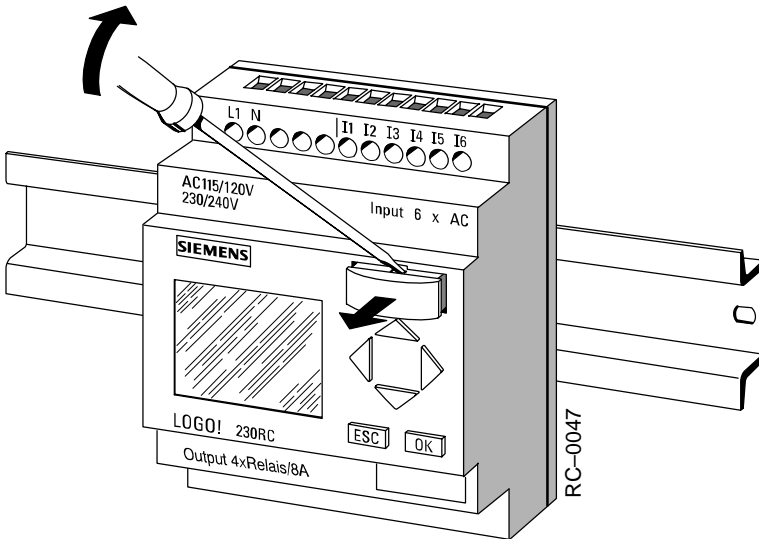
Carefully insert a screwdriver into the slot at the upper end of the program module/card and ease the program module/ card out of the shaft a little.

You can now remove the program module/card:

Inserting the program module/card

The shaft for the program module/card is chamfered at the lower right edge. The program module/card also has a chamfered edge. This prevents you from inserting the program module/card the wrong way around. Insert the program module/card into the shaft until it engages.

5.3 Copying a program from LOGO! to the program module/card

To copy a program to the program module/card, proceed as follows:

1. Insert the program module/card into the shaft.
2. Switch LOGO! to programming mode:
   ◄, ► and **OK** at the same time

```
>Program..
 PC/Card..
 Start
```

3. Move the '>' to "PC/Card":
   ▼
4. Press **OK**. The transfer menu appears.

```
>PC↔LOGO
 LOGO→Card
 Card→LOGO
```

5. Move the '>' to 'LOGO → Card':
   ▼
6. Press **OK**.

LOGO! copies the program to the program module/card.

When LOGO! has finished copying, it automatically returns to the main menu:

```
 Program..
>PC/Card..
 Start
```

The program is now also on the program module/card. You can now remove the program module/card. **Don't forget** to replace the cover.

If there is a power failure while LOGO! is copying, you have to copy the program again once the power has been restored.

5.4 Copying a program from the program module/card to LOGO!

You have a program module/card containing your program. You can copy the program to LOGO! in two ways:

• Automatically when LOGO! starts up (power on)
• Via LOGO!'s PC/Card menu.

Note

Please note that not all programs stored on the modules can be read in all LOGO! variants. If necessary, read Section 5.1 again.

Automatic copying when LOGO! starts up

Proceed as follows:

1. Switch off the power supply to LOGO! (power down)
2. Remove the cover shaft.
3. Insert the program module/card into the appropriate shaft.
4. Switch on the power supply to LOGO! again.

Result: LOGO! copies the program from the program module/card to LOGO!. As soon as LOGO! has finished copying, LOGO! displays the main menu:

```
>Program..
 PC/Card..
 Start
```

Note

Before you switch LOGO! to RUN, you must ensure that the system you are controlling with LOGO! does not represent a source of danger.

1. Move the '>' to Start:
   2 × ▼
2. Press **OK**

Using the PC/Card menu to copy

Read the note about changing the program module/card.

To copy a program from the program module/card to LOGO!, proceed as follows:

1. Insert the program module/card
2. Switch LOGO! to programming mode:
   ◄, ► and **OK** at the same time

```
>Program..
 PC/Card..
 Start
```

3. Move the '>' to "PC/Card":
   ▼

4. Press **OK**. The transfer menu appears.

```
 PC↔LOGO
 LOGO→Card
>Card→LOGO
```

5. Move the '>' to 'Card → LOGO':　　　▲ or ▼
6. Press **OK**.

Result: LOGO! copies the program from the program mod-
ule/card to LOGO!. As soon as LOGO! has finished copy-
ing, LOGO! automatically displays the main menu:

6 LOGO! software

LOGO!Soft and LOGO!Soft Comfort are programming packages for use on PCs. The software has the following functions:

- Offline program generation of your application
- Simulation of your circuit (or your program) on the computer
- Generation and printing of a block diagram of the circuit
- Saving the program to the hard disk or another storage medium
- Program transfer
  - From LOGO! to the PC
  - From the PC to LOGO!

The LOGO! alternative

LOGO! software provides you with an alternative to conventional planning:

1. First you develop your applications at your desk.
2. You simulate the application in your computer and test whether or not it functions properly before the circuit is actually put to use.
3. You print out the entire circuit in a block diagram or in several block diagrams sorted according to outputs.
4. You archive your circuits in your PC file system. In this way you can retrieve a circuit directly if you want to make changes some time in the future.
5. You transfer the program to LOGO! by pressing just a few buttons. Your LOGO! is retooled within a very short space of time.

LOGO!Soft

After program startup, select the LOGO! variant you are working with. This LOGO! variant is displayed on the screen. Using the mouse you can program LOGO! from your desk in the same way you might otherwise have done directly on the system.

LOGO!Soft helps you to create and edit control programs on the PC. The optional 8-language programming software under Windows allows you to create, test and simulate, change, archive and print control programs using your PC.

LOGO!Soft Comfort

LOGO!Soft Comfort makes creating control programs easier, more efficient and more manageable than in the past. Programs are created on the PC using the drag-and-drop technique. You first write the program and then assess which LOGO! variants are required for the completed program.

One particularly user-friendly feature is the offline program simulation facility, which enables the simultaneous status display of multiple special functions and permits extensive documentation of the programs created. This optional program software also provides a comprehensive online help system on CD ROM .

The 6-language LOGO!Soft Comfort software runs as of Windows 95, is server-capable and offers you extensive freedom and maximum ease in creating programs.

Updates & infos

You can download free updates and demo versions of the software from the Internet at the following address:

http://www.ad.siemens.de/logo/html_00/software.htm.

6.1 Possible applications for LOGO! software

System requirements

LOGO!Soft V3.0	LOGO!Soft Comfort V1.0
• IBM-compatible 386 SX16 (486 recommended)	• PC 486 (Pentium 133 recommended)
• 4 MB of RAM (minimum)	• 16 MB of RAM (64 MB recommended)
• 5 MB of free hard disk space (minimum)	• 70 MB of free hard disk space
• Microsoft Windows 3.1	• Microsoft Windows 95/98 or NT4.0
• Screen with VGA resolution (256 color recommended)	• SVGA graphics card, 800x600 (1024x768 recommended)
• Disk drive + mouse	• CD-ROM drive + mouse

Installation and operation

Before installation read the product information documentation and the text files on the disks/CD ROM.

To install the software, simply follow the instructions on the installation program. To start the installation program, proceed as follows (installation on CD-ROM should start automatically):

1. Select SETUP.EXE and start it:
   - In Windows 3.1 e.g. via the File Manager
   - In Windows 95/98 and Windows NT 4.0 either via **Start▸Run** and entering: [Drive]:\Setup or via Windows Explorer
2. Follow the instructions in the installation program.

The best way to find out how to use the software is by working with it on your computer. If you have any questions, use the software online help system.

The next steps

In the next step we will show you how to connect LOGO! to a PC. Omit this step, if you currently only have the software available.

6.2 Connecting LOGO! to a PC

Connecting a PC cable

To connect LOGO! to a PC, you need the LOGO! PC cable (order no. 6ED1 057–1AA00–0BA0).

Remove the cover or the program module/card on your LOGO! and connect the cable there. Connect the other end of the cable to the serial port of your PC.
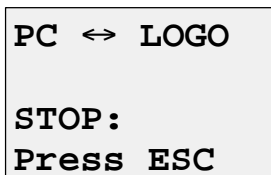
Switch LOGO! to the operating mode PC↔LOGO

There are two ways to connect the PC and LOGO!. LOGO! is either switched to transfer mode when it is on or automatically when the LOGO! power supply is switched on and the transfer cable is in place.

To switch LOGO! to PC ↔ LOGO mode, proceed as follows:

1. Switch LOGO! to programming mode:
   Press ◄, ► and **OK** at the same time
2. Select 'PC/Card':
   ▼ or ▲
3. Press **OK**
4. Select 'PC ↔ LOGO':
   ▼ or ▲
5. Press **OK**

LOGO! is now in PC ↔ LOGO mode and the following display appears:

```
PC ↔ LOGO


STOP:
Press ESC
```

To switch LOGO! automatically to PC ↔ LOGO mode, proceed as follows:

1. Switch off the power supply to LOGO!
2. Remove the cover or the program module/card and connect the cable there.
3. Switch the power back on.

LOGO! automatically goes into PC ↔ LOGO mode.

The PC can now access LOGO!. You can find out how to do this on the online help system of the LOGO! software.

You can interrupt the connection to the PC using ESC on LOGO!.

6.3 Transfer settings

To transfer programs between the PC and LOGO!, you must make certain settings in the LOGO! software. You can make these settings using the menu of the software you are using.

LOGO!Soft

- **Select LOGO!**: In the case of LOGO!Soft you must set the variant that is connected to the PC. The programs are only interchangeable if the two variants match up.
- **Set up link**: Specify the serial interface of the PC via which data is to be exchanged with LOGO!.
- **PC ! LOGO!**: Use this to transfer a program you have generated in LOGO!Soft to LOGO!.
- **LOGO! ! PC**: Use this to transfer a program you have generated in LOGO! to LOGO!Soft.

LOGO!Soft Comfort

- **Specify which LOGO!**: LOGO!Soft Comfort works out which LOGO! variant you will need to use the written program.
- **Options ! Interface**: You can enter the serial port to which LOGO! is connected. You can also find out the correct port automatically (each port is checked by the program for a connected LOGO!).
- **Transfer: PC ! LOGO!**: Use this to transfer a program created in LOGO!Soft Comfort to LOGO!.
- **Transfer: LOGO! ! PC**: Use this to transfer a program created with LOGO! to LOGO!Soft Comfort.

7 Applications

To give you an idea of the kind of situations in which you can use LOGO!, we have compiled a number of application examples. We have included the circuit diagram of the original solution for each example and compared it with the solutions using LOGO!.

You can find solutions for the following tasks:

Note

The LOGO! applications are provided free of charge to our customers. The examples they contain are not binding and are included to provide general information on how LOGO! can be used. Customer-specific solutions may be different.

The user is responsible for ensuring that the system operates properly. We would also draw your attention to the need to comply with any applicable local standards and system-related installation regulations.

Errors are excepted and the right to make changes reserved.

You can find these applications and tips for further applications on the Internet at:
http://www.ad.siemens.de/logo

7.1 Stairway, hall or corridor lighting

7.1.1 Demands on stairwell lighting

The basic requirements for a stairwell lighting system in an apartment block are as follows:

- When someone is using the stairs, the stairwell lights should be on.
- If no-one is in the stairwell, the lights should go out in order to save energy.

7.1.2 Previous solution

Conventionally, there have been two ways to control such a lighting system as follows:

- By means of a current impulse relay
- By means of automatic stairway lighting

The wiring for the two lighting systems is identical.
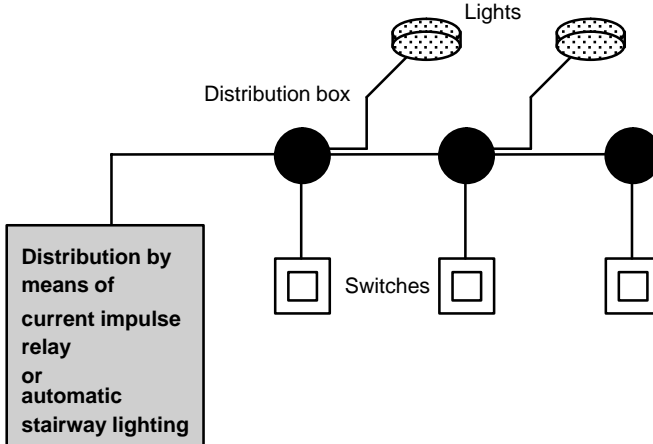
Components used

- Key
- Automatic lighting device or current impulse relay

Lighting system with a current impulse relay

When a current impulse relay is used, the lighting system behaves as follows:

• When any switch is pressed: the lighting is switched on
• When any switch is pressed again: the lighting is switched off

Disadvantage: people frequently forget to switch the lights off.

Lighting system with an automatic lighting device

When an automatic device is used, the lighting system behaves as follows:

• When any switch is pressed: the lighting is switched on
• When the preset time has elapsed, the lights switch off automatically.

Disadvantage: The lighting cannot be switched on for an extended period of time (e.g. for cleaning purposes). The permanent on switch is usually on the stairwell lighting timer unit which may be difficult or impossible to gain access to.

7.1.3 Lighting system with LOGO!

You can use a LOGO! module to replace the stairwell lighting timer or the current impulse relay. And you can implement both functions (timed off-delay and current impulse relay) in a single unit. What is more, you can incorporate extra functions without making any alterations to the wiring. Here are some examples:

• Current impulse relay with LOGO!
• Automatic stairway lighting system with LOGO!
• LOGO! as a multifunctional switching system with the following functions:
    – Light on: Press switch
      (Light switches off after the set time elapses)
    – Permanent light on: Press switch twice
    – Light off: Press switch for 2 seconds

Wiring of the lighting system with LOGO! 230RC
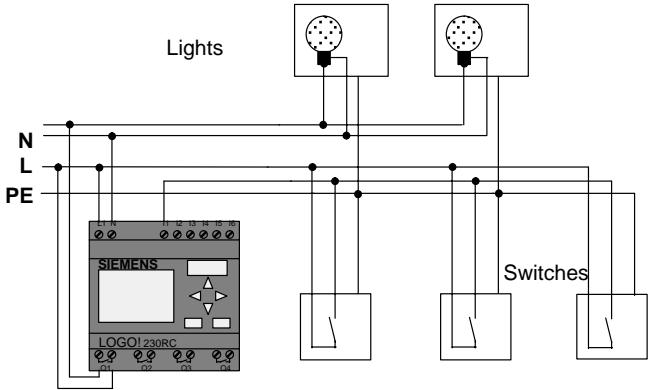
The external wiring for the lighting system using a LOGO! module is no different from the conventional method of wiring a corridor or stairwell lighting system. Only the automatic lighting timer/current impulse relay is replaced. Additional functions are entered directly in LOGO!.

Current impulse relay with LOGO!
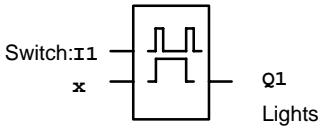
In the event of a gate pulse at input I1, output Q1 switches over.

Automatic stairway lighting system with LOGO!
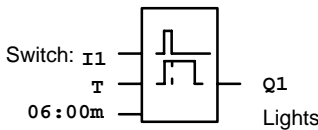
In the event of a gate pulse at input I1, output Q1 switches on and remains on for 6 minutes.

Multifunctional switch with LOGO!
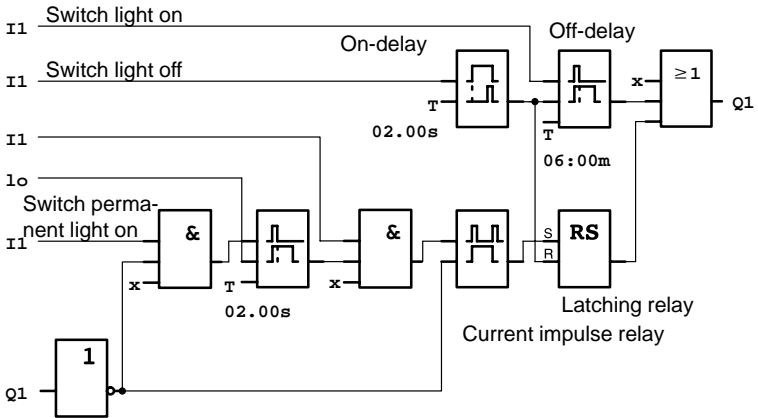
The diagram shows the circuit for an input with an associated output.

This switch has the following options:

- **When the switch is pressed:** The light is switched on and goes off again after the set time of 6 minutes (T=06:00m) has elapsed (off-delay)
- **When the switch is pressed twice:** The light is switched on permanently
  (the latching relay is set via the current impulse relay).
- **When the switch is pressed for 2 seconds:** The light is switched off (on-delay switches the light off; both the permanent light and the normal light; this branch of the circuit is therefore used twice)

You can enter these circuits several times for the remaining inputs and outputs. Instead of using 4 automatic stairway lighting systems or 4 current impulse relays, you thus use only a single LOGO!. However, you can also use the free inputs and outputs for completely different functions.

7.1.4 Special features and enhancement options

Features such as the following are available for adding functions or saving energy:

- You can have the light flash before it goes off automatically.
- You can integrate various central functions:
    - Central off
    - Central on (panic button)
    - Control of all lights or individual circuits by a daylight control switch
    - Control by the integrated time switch (e.g. permanent light only until 24.00 hours; no enabling at certain times)
    - Automatic switching off of the permanent light after a preset time has elapsed (e.g. after 3 hours)

7.2 Automatic door

You often find automatic door control systems at the entrance to supermarkets, public buildings, banks, hospitals etc.

7.2.1 Demands on an automatic door

- When somebody approaches, the door must open automatically.
- The door must remain open until there is nobody in the doorway anymore.
- If there is nobody in the doorway anymore, it must close automatically after a short time.
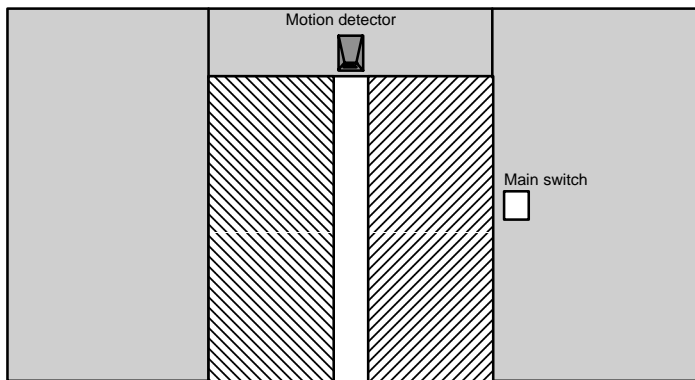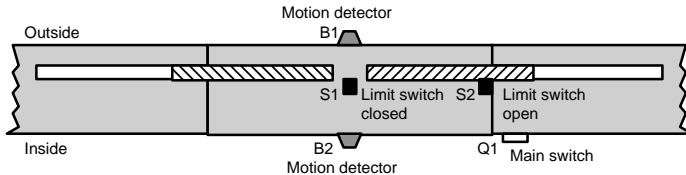
The door is generally driven by a motor with a safety clutch. This prevents people from being caught or injured in the door. The control system is connected to the mains via a main switch.

7.2.2 Previous solution
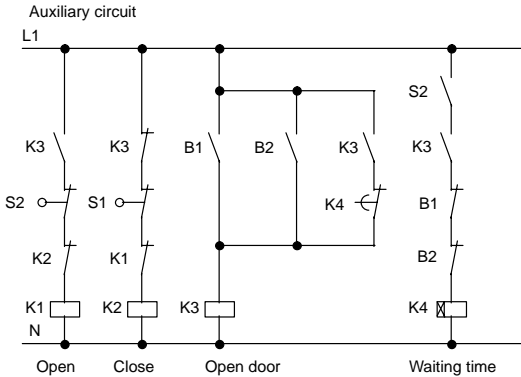
As soon as one of the motion detectors B1 or B2 registers somebody's presence, the door is opened by K3.

If the two motion detectors detect nothing for a minimum period, K4 enables the close operation.

7.2.3 Door control system with LOGO!

LOGO! allows you to considerably simplify the circuit. You need only connect the motion detectors, the limit switches and the master contactors to LOGO!.

Wiring of the door control system with LOGO! 230RC
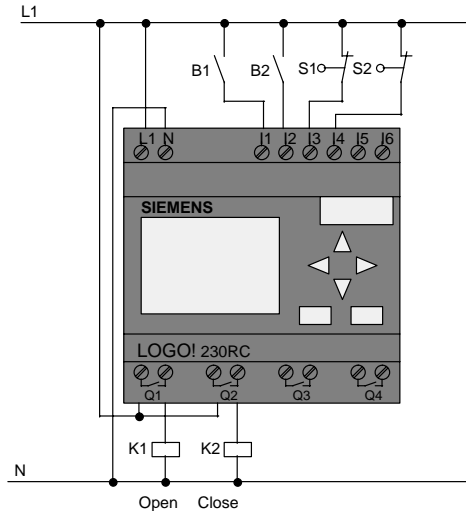
Components used

- K1   Master contactor *Open*
- K2   Master contactor *Close*
- S1 *(NC contact)*   Limit switch *Closed*
- S2 *(NC contact)*   Limit switch *Open*
- B1 *(NO contact)*   Infrared motion detector *Outside*
- B2 *(NO contact)*   Infrared motion detector *Inside*

Diagram of the door control system with LOGO!
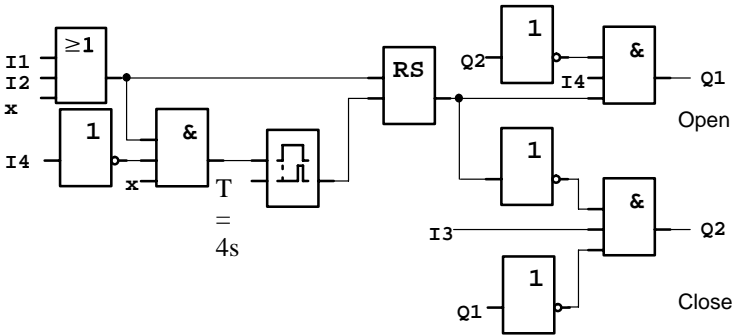
This is what the functional block diagram that corresponds to the circuit diagram of the conventional solution looks like.

You can simplify this circuit if you make use of LOGO!'s functions. You can use the off-delay to replace the latching relay and on-delay. The following function block diagram illustrates this simplification:
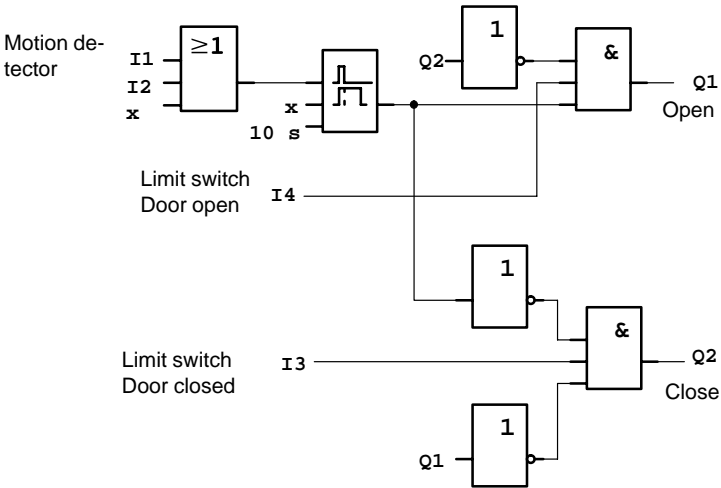
7.2.4 Special features and enhancement options

Functionality and user friendliness can be improved in the following ways, for example:

- You can connect an additional control switch: Open – Automatic – Closed (O-A-C)
- You can connect a buzzer to one of LOGO!'s outputs to indicate when the door is about to close.
- You can include time and direction-dependent enabling of door opening (so that it only opens during shop opening hours or only from the inside to the outside after closing time).

7.2.5 Enhanced LOGO! 230RC solution

Wiring of the enhanced LOGO! solution
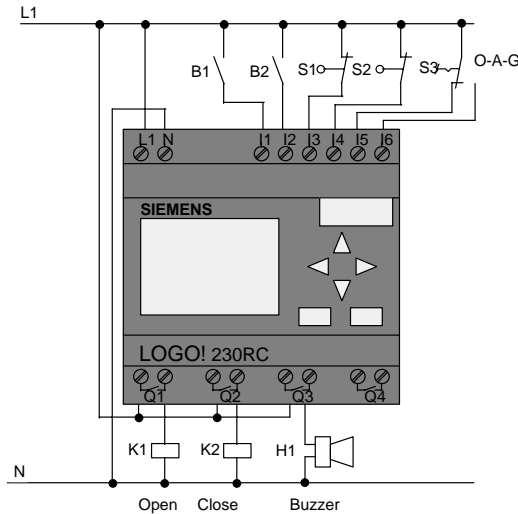
Functional block diagram of the enhanced LOGO! solution

No1:
Day= Mo..Fr
On = 09:00
Off =18:00
No2:
Day= Sa
On = 08:00
Off =13:00

Detecting motion
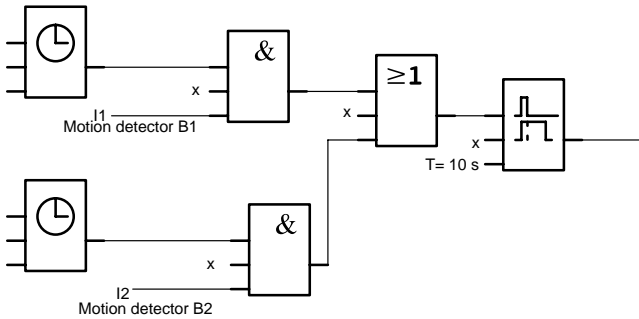
No1:
Day= Mo..Fr
On = 09:00
Off =19:00
No2:
Day= Sa
On = 08:00
Off =14:00

I1
Motion detector B1

x

&

x

≥1

x

T= 10 s

x

&

I2
Motion detector B2

Motor for opening

Close output   Q2

Limit switch
Door open   I4

Control switch
Door open   I5

1

&

x

≥1

Q1
Open

Motor for closing

Limit switch
Door closed   I3

Open output   Q1
Motion detector B1   I1
Motion detector B2   I2

Control switch
Close door   I6

1

&

≥1

x

≥1

Q2
Close
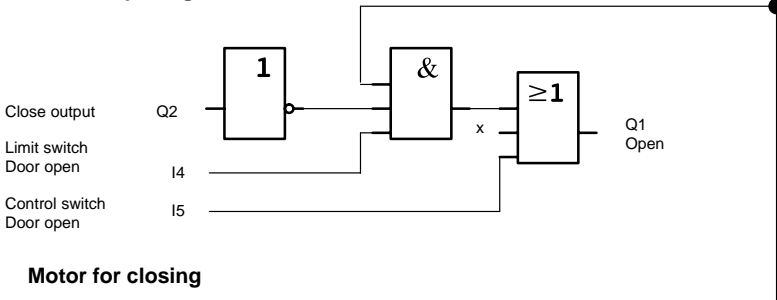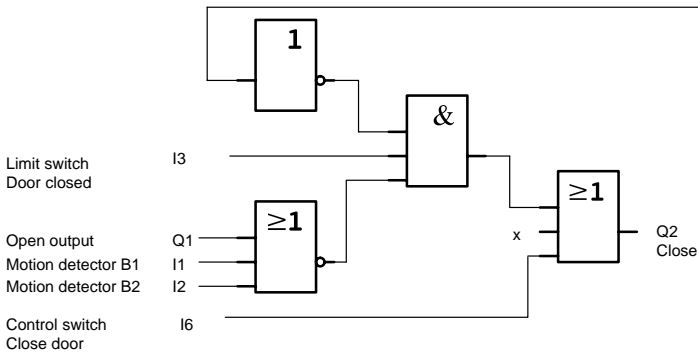
Detecting motion

> During business hours, motion detector B1 opens the door as soon as somebody wants to enter the shop from outside. Motion detector B2 opens the door if somebody wants to leave the shop.

> After closing time, motion detector B2 continues to open the door for 1 hour so that customers can leave the shop.

Motor for opening

> Output Q1 is switched on and opens the door when the following occurs:
> - The control switch at I5 is operated (the door is to be constantly open) or
> - The motion detectors indicate that somebody is approaching the door and
> - The door is not yet fully open (limit switch at I4).

Motor for closing

> Output Q2 is switched on and closes the door when the following occurs:
> - The control switch at I6 is operated (the door is to be constantly closed) or
> - The motion detectors indicate that there is nobody near the door and
> - The door is not yet fully closed (limit switch at I3).

Buzzer

> Connect the buzzer to output Q3. The buzzer sounds for a short time (in this case 1 second) when the door is closed. Enter the following circuit at Q3 in the block diagram:
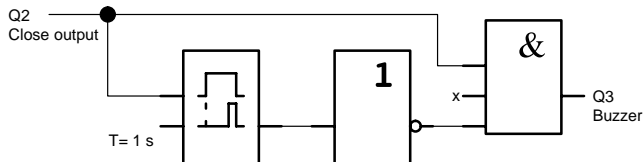
7.3 Ventilation system

7.3.1 Demands on a ventilation system

The purpose of a ventilation system is either to bring fresh air into a room or to extract stale air from a room. Consider the following example:

- The room contains an extractor fan and a fresh-air fan.
- Both fans are monitored by a flow sensor.
- The pressure in the room must not be allowed to rise above atmospheric pressure.
- The fresh-air fan must only be switched on provided reliable functioning of the extractor fan is signaled by the flow sensor.
- A warning lamp indicates if one of the fans fails.

The circuit diagram for the previous solution is as follows:

Auxiliary circuit

Stale air   Fresh air   Operation Fault
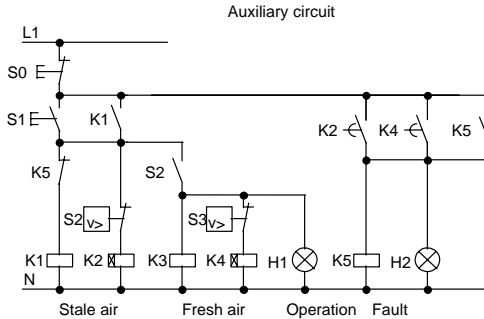
The fans are monitored by flow sensors. If, after a short delay, no air flow is registered, the system is switched off and a fault reported. Acknowledge this by pressing the stop switch.

Monitoring the fans requires an analyzer circuit with several switching devices in addition to the flow sensors. The analyzer circuit can be replaced by a single LOGO! module.

Wiring of the ventilation system with LOGO! 230RC
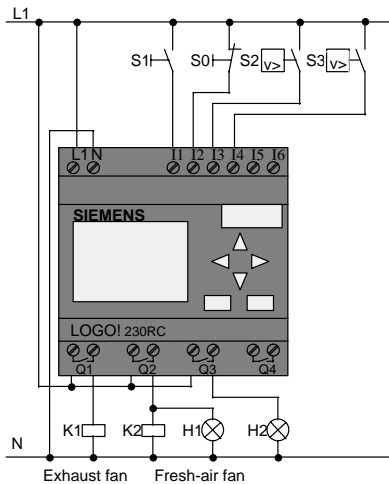
Exhaust fan   Fresh-air fan

Components used

- K1   Master contactor
- K2   Master contactor
- S0 *(NC contact)*   Stop switch
- S1 *(NO contact)*   Start switch
- S2 *(NO contact)*   Flow sensor
- S3 *(NO contact)*   Flow sensor
- H1   Indicator lamp
- H2   Indicator lamp

Block diagram of the LOGO! solution

The block diagram of the ventilation control system with LOGO! is as follows:

7.3.2   Advantages of using LOGO!

If you use a LOGO! module you require fewer switching devices. That saves you installation time and space in the control cabinet. Under certain circumstances it may even allow you to use a smaller control cabinet.

Additional options when using LOGO!

- The free output (Q4) can be used as a potential-free signaling contact in the event of a fault or a power failure.
- It is possible to stagger the switching-off of the fans.

These functions can be implemented without additional switching devices.

Functional block diagram of the enhanced LOGO! solution

The fans at Q1 and Q2 are switched on and off as shown in the following circuit:

You can also generate a message via output Q4:

```
        ┌─────┐
        │  1  │
Q3   ───┤     ├○─ Q4
Fault   └─────┘   Message
```
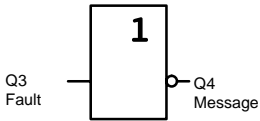
The relay contacts of output Q4 are always closed when the system is running. Relay Q4 does not release unless there is a power failure or a fault in the system. This contact can be used for teleindication, for example.

7.4 Industrial gate

Safety pressure bar

Warning light
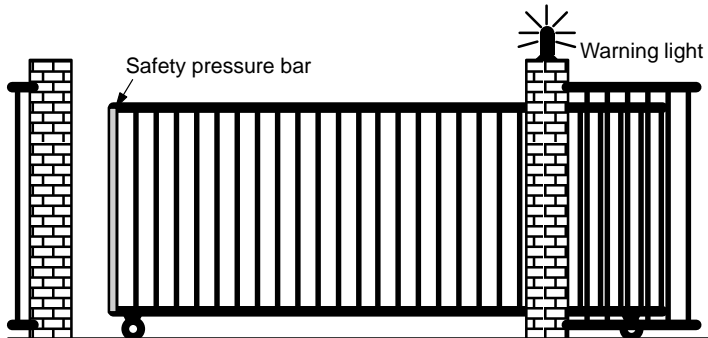
There is often a gate at the entrance to a company's premises. This is only opened to let vehicles in and out.

The gate is controlled by the gateman.

7.4.1 Demands on the gate control system

- The gate is opened and closed by operating a switch in the gatehouse. The gateman is able to monitor the operation of the gate at the same time.
- The gate is normally fully opened or closed. However, movement of the gate can be halted at any time.
- A flashing light is activated 5 seconds before the gate begins to move and continues for as long as the gate is in motion.
- A safety pressure bar ensures that nobody gets injured and nothing gets trapped or damaged when the gate closes.

7.4.2   Previous solution

Various kinds of control systems are used to operate auto-
matic gates. The circuit diagram shows *one* possible gate
control circuit.
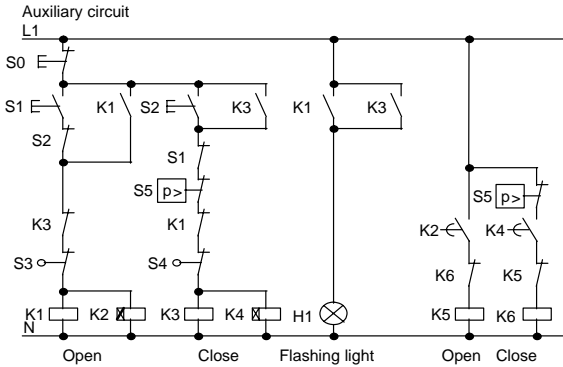
Wiring of the gate control system with LOGO! 230RC
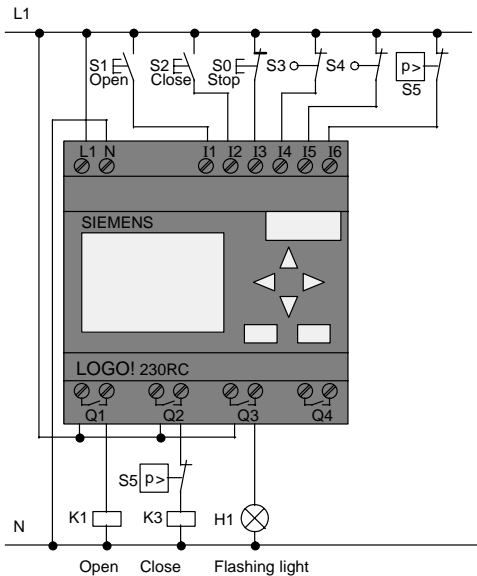
Components used

- K1   Master contactor
- K2   Master contactor
- S0 *(NC contact)*   Stop switch
- S1 *(NO contact)*   Open switch
- S2 *(NO contact)*   Close switch
- S3 *(NC contact)*   Open position switch
- S4 *(NC contact)*   Closed position switch
- S5 *(NC contact)*   Safety pressure bar

Functional diagram of the LOGO! solution
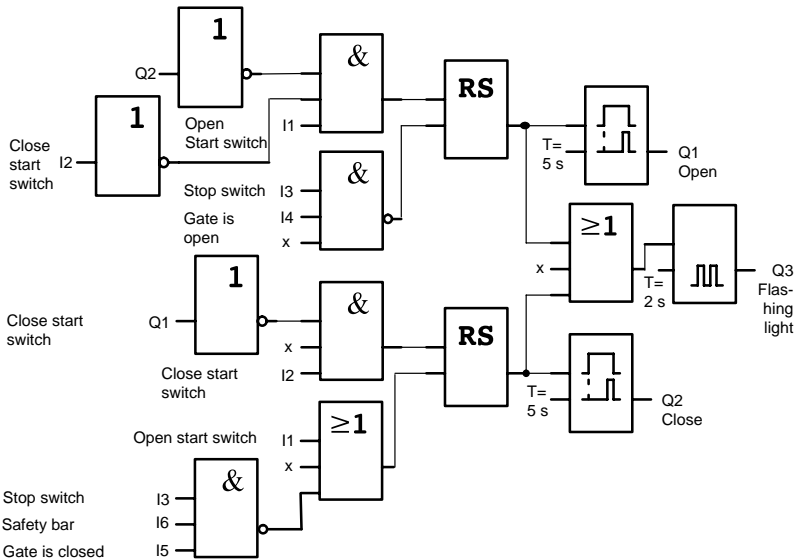
The open and close start switches move the gate in the appropriate direction provided it is not already moving in the other direction. Movement of the gate is halted by the stop switch or the relevant limit switch. The gate is also prevented from closing by the safety bar.

7.4.3 Enhanced LOGO! solution

In our enhanced solution the gate will automatically open again when the safety bar is operated.
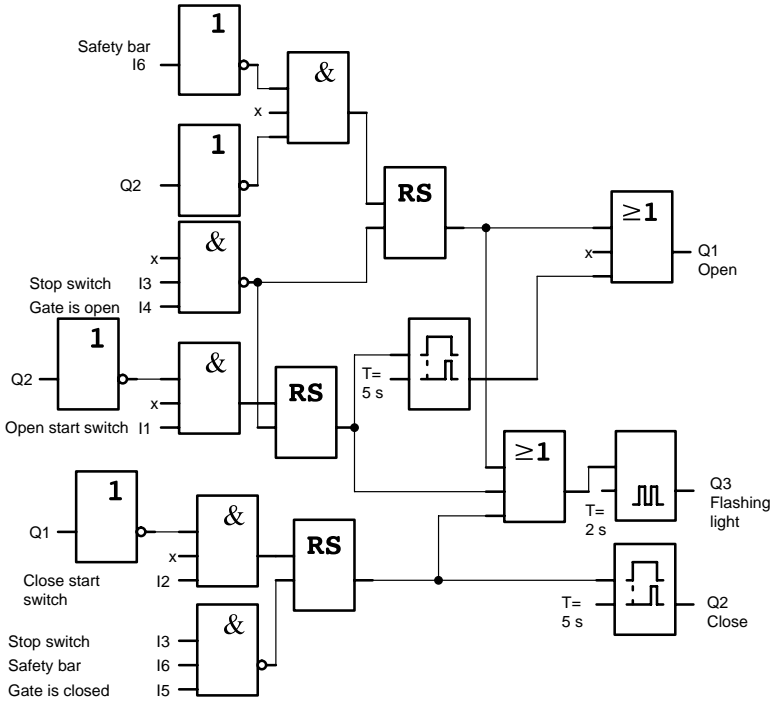
7.5 Centralized activation and surveillance of several industrial gates

1 Safety pressure bar
2 Flashing warning light
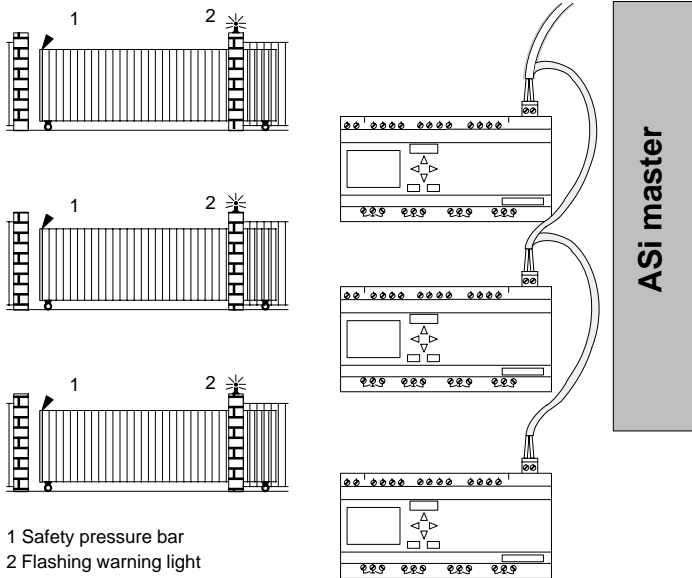
There are often a number of different entrances to a company's premises. Not all gates can always be surveilled directly by a member of staff. They must therefore be able to be surveilled and operated by a gateman who sits in a central control room.

It is also important to ensure that a member of staff can open and close the gate directly at the gate.

A LOGO!230RCLB11 is used for each gate. The modules are linked to each other and an ASi master by means of the ASi bus.

In this chapter, we will describe the gate control system used for a gate. All the other gate control systems are identical.

7.5.1 Demands on the gate control system

- Each gate is opened and closed by means of a pull-cord switch. The gate is normally fully opened or closed.
- Each gate can also be opened and closed using switches at the gate.
- The ASi bus connection enables the gateman to open and close the gate from the gatehouse. The state GATE OPEN or GATE CLOSED is indicated in the gatehouse.
- A flashing light is activated 5 seconds before the gate begins to move and continues for as long as the gate is in motion.
- A safety pressure bar ensures that nobody gets injured and nothing gets trapped or damaged when the gate closes.

Wiring of the gate control system with LOGO! 230RCLB11
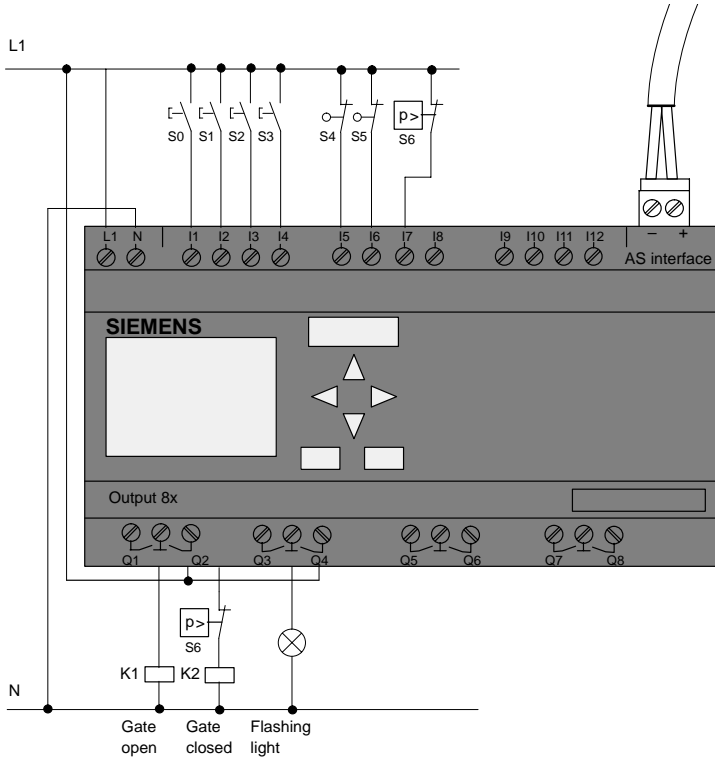
Components used

- K1   Master contactor opening
- K2   Master contactor closing
- S0 *(NO contact)*   OPEN pull-cord switch
- S1 *(NO contact)*   CLOSE pull-cord switch
- S2 *(NO contact)*   OPEN switch
- S3 *(NO contact)*   CLOSE switch
- S4 *(NC contact)*   OPEN GATE position switch
- S5 *(NC contact)*   CLOSE GATE position switch
- S6 *(NC contact)*   Safety pressure bar

Higher-level control system

- Qa1   GATE OPEN position switch
- Qa2   GATE CLOSED position switch
- Ia1   External OPEN GATE switch
- Ia2   External CLOSE GATE switch

Functional diagram of the LOGO! solution
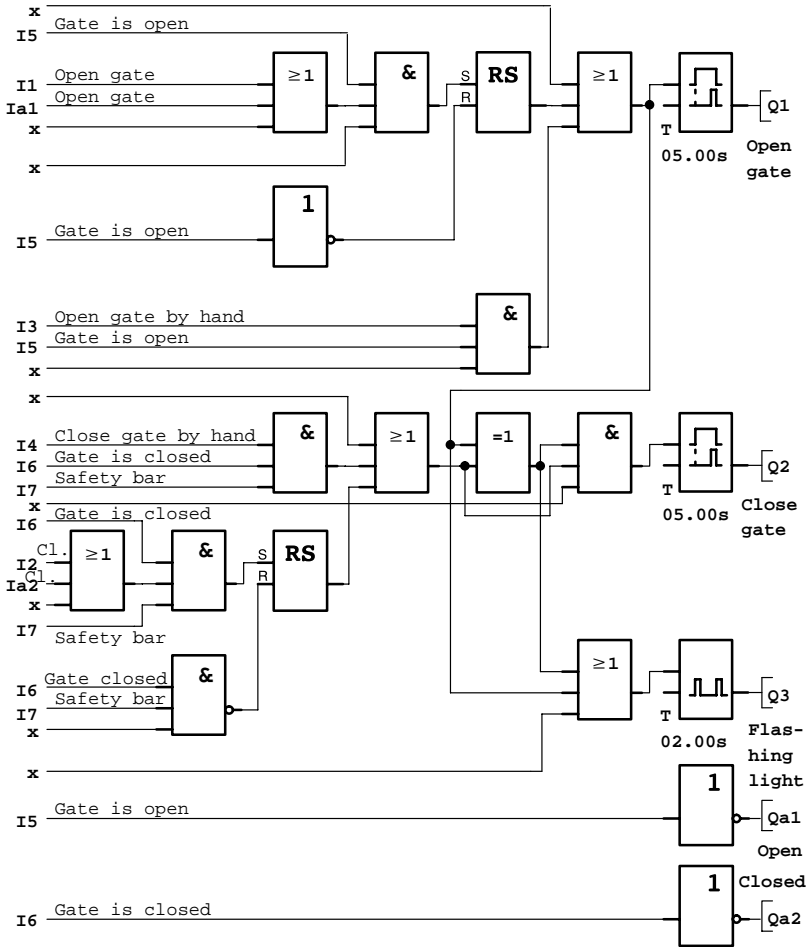
The OPEN gate and CLOSE gate start switches move the gate in the appropriate direction provided it is not already moving in the other direction. Movement of the gate is halted by the relevant limit switch. The gate is also prevented from closing by the safety bar.

7.6 Fluorescent lamps
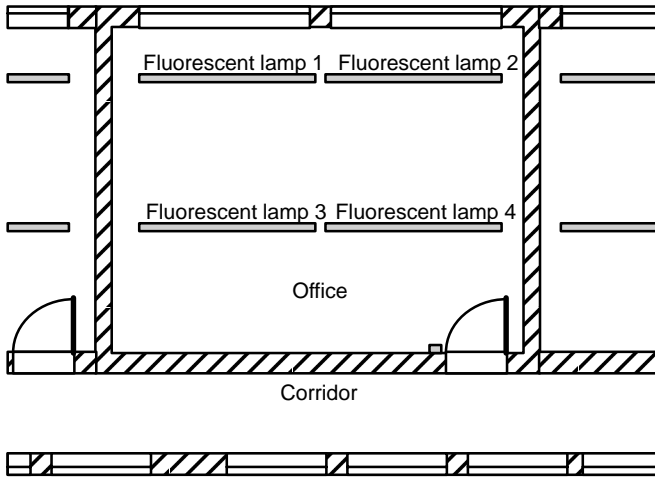

When lighting systems are planned in companies, the type and number of lamps used depends on the level of lighting required. For reasons of cost efficiency, fluorescent lamps arranged in rows of tubes are often used. They are subdivided into switching groups according to how the room is used.

7.6.1 Demands on the lighting system

- The lamps are switched on and off locally.
- If there is sufficient natural light, the lamps on the window side of the room are automatically switched off by means of a brightness-sensitive switch.
- The lights are switched off automatically at 20.00.
- It must be possible at all times to switch the lights on and off locally.

7.6.2 Previous solution
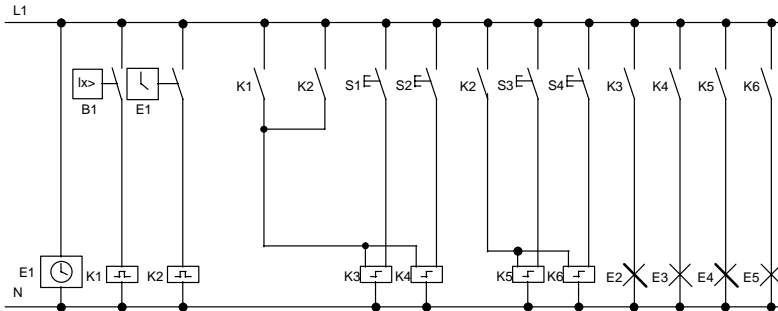
The lights are operated by means of a current impulse relay controlled by switches at the door. Independently of this, they are reset by the time switch or by the brightness-sensitive switch via the *central off* input. The switching-off commands must be cut by interval time-delay relays so that it is still possible to switch the lights on and off locally after they have been switched off centrally.

Components required:

- Switches S1 to S4
- Daylight control switch B1
- Time switch E1
- Interval time-delay relays K1 and K2
- Remote-control switches with central off K3 to K6

Disadvantages of the previous solution

- To implement the required functions, a large amount of circuitry is necessary.
- The large number of mechanical components means that considerable wear and high maintenance costs can be expected.
- Functional changes are costly to implement.

7.6.3 Fluorescent lamp control with LOGO! 230RC

Fluorescent lamp 1   Fluorescent lamp 2   Fluorescent lamp 3   Fluorescent lamp 4
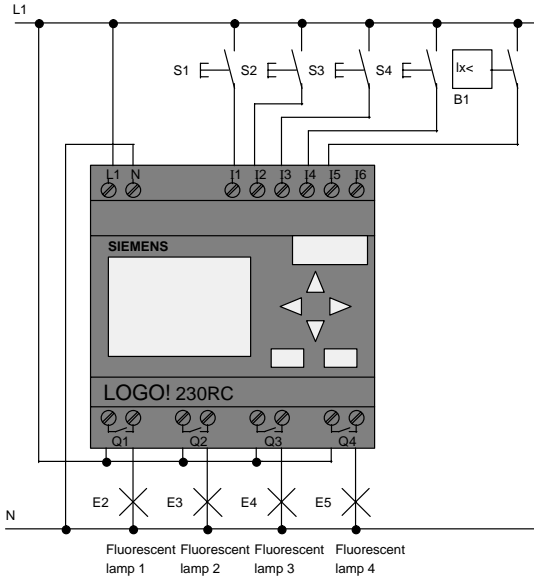
Components used

- S1 to S4 *(NO contact)*   Key
- B1 *(NO contact)*   Daylight control switch

Functional diagram of the LOGO! solution
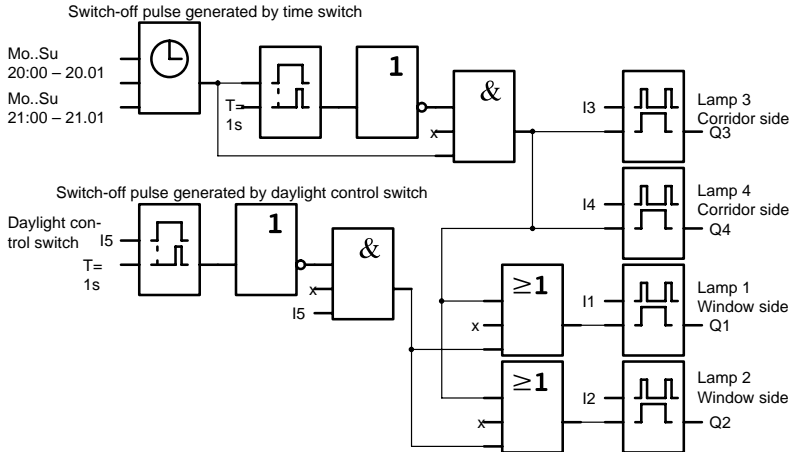
Advantages of the LOGO! solution

- You can connect the lamps directly to LOGO! provided the switching capacity of the individual outputs is not exceeded. In the case of greater capacities, you should use a power contactor.
- Connect the brightness-sensitive switch directly to one of LOGO!'s inputs.
- You don't need a time switch since this function is integrated in LOGO!.
- The fact that fewer switching devices are required means that you can install a smaller sub-distribution unit and thus save space.
- Fewer devices are required
- The lighting system can be easily modified.
- Additional switching times can be set as required (staggered switch-off pulses at the end of the day).
- The brightness-sensitive switch function can easily be applied to all lamps or an altered group of lamps.

7.7 Rainwater pump

Nowadays, private households are increasingly making use of rainwater alongside the mains domestic water supply. This saves money and helps protect the environment. Rainwater can be used for the following, for example:

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

The sketch below illustrates how such a rainwater supply system works:

Rainwater inlet

K4 Mains water inlet

Pressure switch

S1

Control in the distribution box

Pump

M1 Pressure tank

Rainwater pipe

Rainwater collecting tank

S2 Mains water inlet off

S3 Mains water inlet on
Run-dry prevention off

S4 Run-dry prevention on
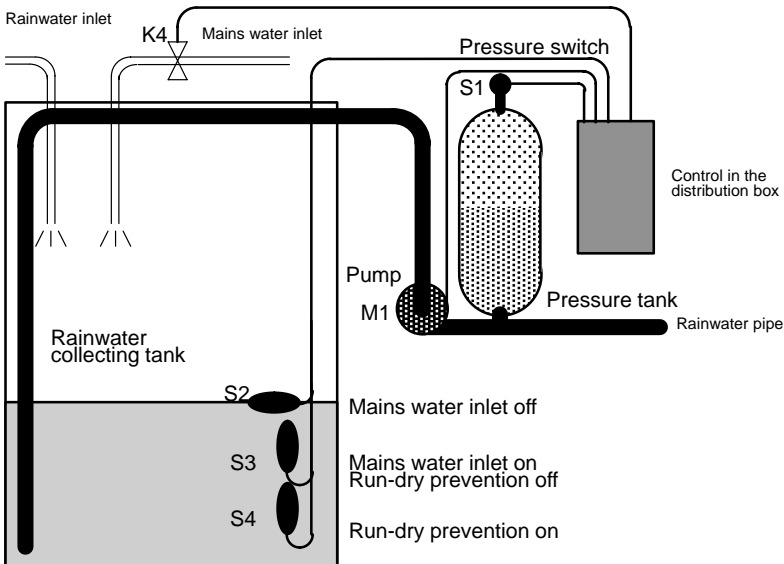
The rainwater is collected in a large water butt. From the water butt it is pumped by a pumping station into a piping system provided for it. From there it can be drawn off in the same way as the normal household water supply. If the water butt should run dry, it can be topped up with mains water.

7.7.1 Demands on the control system of a rainwater pump

- The system must be capable of supplying water at all times. If necessary, the control system must switch over to mains water if the rainwater runs out.
- The system must not allow rainwater to enter the mains supply when switching over to mains water.
- The pump cannot be switched on if there is not enough water in the rainwater butt (run-dry prevention system).

7.7.2 Previous solution
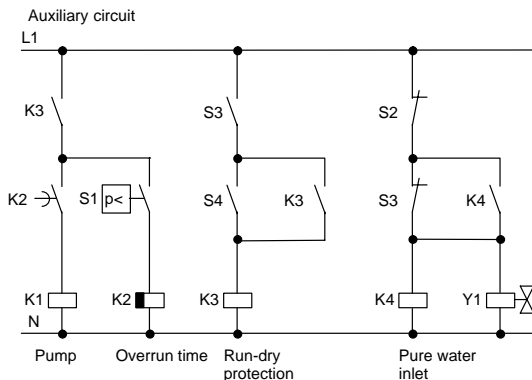
The pump and a solenoid valve are controlled by means of a pressure switch and 3 float switches that are fitted in the rainwater butt. The pump must be switched on when the pressure in the cylinder falls below the minimum level. Once the operating pressure has been reached, the pump is switched off again following a short overrun period of a few seconds. The overrun time prevents the water pump continuously cutting in and out if water is drawn off for any length of time.

7.7.3   **Rainwater pump with LOGO! 230RC**
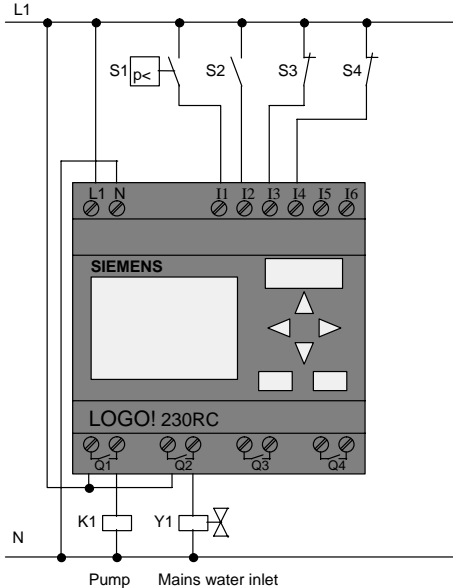
Apart from LOGO!, you only need the pressure switch and the float switches to control the pump. If you are using a 3-phase AC motor, you must use a master contactor for switching the pump. On systems using single-phase AC pumps, you must fit a contactor if the AC pump requires a higher current than can be switched by the output relay Q1. The power of a solenoid valve is so low that it can normally be controlled directly.

- K1   Master contactor
- Y1   Solenoid valve
- S1 *(NO contact)*   Pressure switch
- S2 *(NO contact)*   Float switch
- S3 *(NC contact)*   Float switch
- S4 *(NC contact)*   Float switch

Block diagram of the LOGO! solution
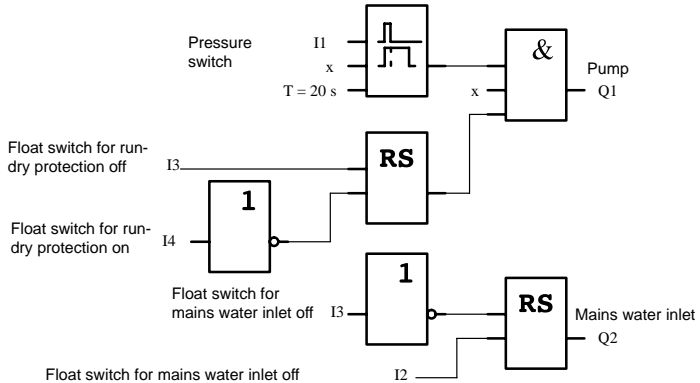
7.7.4 Special features and enhancement options

The function diagram shows how you can connect up the control system for the pump and the solenoid valve. Its layout corresponds to that of the current flow diagram. You also have the possibility of incorporating additional functions for specific applications that would only be possible with the inclusion of extra equipment if you were using conventional technology, e.g.:

- Enabling the pump at specific times
- Indication of imminent or existing shortage of water
- Reporting of system faults

7.8 Other applications

In addition to the above applications, we have also put a selection of 23 further potential applications on the Internet (updated June 1999) at www.ad.siemens.de/logo/html_00/einsatz.htm.

The following are some of the examples you can find there:

- Watering greenhouse plants
- Control of conveyor systems
- Control of a bending machine
- Shop window lighting
- Bell system (e.g. in a school)
- Parking lot surveillance
- Exterior lighting
- Control system for shutters
- Interior and exterior lighting for an apartment block
- Control system for a cream stirrer
- Sports hall lighting
- Equal distribution of 3 loads
- Sequence control system for cable-welding machines with large cross-sections
- Step switch (e.g. for fans)
- Boiler sequence control
- Control system for several pump sets with centralized operation
- Cutting device (e.g. for detonating fuses)
- Monitoring length of use (e.g. in solar energy system)
- Intelligent foot buttons (e.g. to select speeds)
- Control of an elevating platform
- Impregnation of textiles – heating and conveyor control system
- Silo-filling system

And many more.

You can also find descriptions and the relevant circuit diagrams of the applications on the Internet. You can read these *.pdf files with Adobe Acrobat Reader. And if you have installed the LOGO!Soft or LOGO!Soft Comfort programming software on your PC, you can simply download the relevant circuit diagrams at the touch of a button, adapt them to your requirements, transfer them into LOGO! via the PC cable and start to use them.

Advantages of using LOGO!

It is worth using LOGO! particularly when you:

- Can replace a number of auxiliary switching devices with the integrated functions of LOGO!.
- Want to save yourself wiring and installation work (because the wiring is done in LOGO!).
- Want to reduce the space required for components in the control cabinet/distribution box. You may be able to use a smaller control cabinet/distribution box.
- Can add or change functions subsequently without having to install an additional switching device or change the wiring.
- Have to provide your customers with additional functions for their domestic or building installation. Here are some examples:
  - Home security: You can program LOGO! to switch a lamp on regularly or open and close your shutters while you are on holiday.
  - Heating system: You can program LOGO! to run the circulation pump only when water or heat is really required.
  - Cooling systems: You can program LOGO! to thaw your cooling systems automatically on a regular basis to save energy costs.
  - You can illuminate aquaria and terraria automatically on a time-dependent basis.

You can also:

- Use commercially available switches and buttons, which makes it easy to integrate in the installation.
- Connect LOGO! directly to your domestic installation due to its integrated power supply.

Do you have any suggestions?

There are definitely more ways to use LOGO!. If you know of one, why not write to us? We will collect all the suggestions we receive, and we intend to pass on as many as we can. So drop us a line – no matter how unusual or simple your LOGO! circuit is! We would be delighted to receive your suggestions.

Write to:

Siemens AG
A&D AS MVM – LOGO!
Postfach 48 48
D-90327 Nürnberg

A Technical specifications

A.1 General technical specifications

Criterion	Complies with	Values
Dimensions WxHxD in mm		72 x 90 x 55 with installation mechanism: 72 x 90 x 59
Weight		Approx. 190 g
Installation		On 35 mm DIN rail width of 4 modules
Dimensions WxHxD in mm (LOGO!...L...)		126 x 90 x 55 with installation mechanism: 126 x 90 x 59
Weight (LOGO!...L...)		Approx. 360 g
Installation (LOGO!...L...)		On 35 mm DIN rail width of 7 modules
Climatic environmental conditions		
Ambient temperature	Cold: IEC 68-2-1 Heat: IEC 68-2-2[*]	
Horizontal installation		0 ... 55 ˚C
Vertical installation		0 ... 55 ˚C
Storage/transport		-40 ˚C ... +70 ˚C
Relative humidity	IEC 68-2-30	From 5 to 95% no condensation
Air pressure		795 ... 1080 hPa
Pollutants	IEC 68-2-42	SO_2 10 cm^3 /m^3, 4 days
	IEC 68-2-43	H_2S 1 cm^3 /m^3, 4 days
Mechanical environmental conditions		
Type of protection		IP20

[*]IEC 68 includes VDE 0631

Criterion	Complies with	Values
Vibrations:	IEC 68-2-6	10 ... 57 Hz (constant amplitude 0.15 mm)
		57 ... 150 Hz (constant acceleration 2 g)
Shock	IEC 68-2-27	18 shocks (Half-sine 15g/11ms)
Drop	IEC 68-2-31	Drop height 50 mm
Free fall (packaged)	IEC 68-2-32	1 m
Electromagnetic compatibility (EMC)		
Electrostatic discharge	IEC 801-2 Severity 3	8 kV air discharge 6 kV contact discharge
Electromagnetic fields	IEC 801-3	Field strength 10 V/m
Interference suppression	EN 55011	Limit class B group 1 Limit class for ASi operation
EMC emitted interference	EN 50081-2	
Interference immunity	EN 50082-2	
Burst pulses	IEC 801-4 Severity 3	2 kV (supply and signal lines) B11 variants: to *ASi Complete Specification V 2.0 dated 27.11.95*
Energy carriers Single pulse (surge) (applies only to LOGO! 230....)	IEC 801-5 Severity 2	0.5 kV (power lines) symmetrical 1 kV (power lines) asymmetrical
Information on IEC − / VDE − safety		
Measurement of clearance and creepage distance	IEC 664, IEC 1131, EN 50178 draft 11/94 UL 508, CSA C22.2 No 142 Also VDE 0631 for LOGO! 230R/RC	Fulfilled
Insulation strength	IEC 1131	Fulfilled

A.2 Technical specifications: LOGO! 230...

	LOGO! 230RC LOGO! 230RCo	LOGO! 230RCL LOGO! 230RCLB11
Power supply		
Input voltage	115/230 V AC	115/230 V AC
Permissible range	85 ... 250 V AC	85 ... 250 V AC
Permissible mains frequency:	47 ... 63 Hz	47 ... 63 Hz
Power consumption • 115 V AC • 230 V AC	 10 ... 30 mA 10 ... 20 mA	 15 ... 65 mA 15 ... 40 mA
Voltage failure bridging • 115 V AC • 230 V AC	 Typically 10 ms Typically 20 ms	 Typically 10 ms Typically 20 ms
Power loss at • 115 V AC • 230 V AC	 1.1 ... 3.5 W 2.3 ... 4.6 W	 1.7 ... 7.5 W 3.4 ... 9.2 W
Clock buffering at 25 °C	Typically 80 h	Typically 80 h
Accuracy of the real-time clock	Max. ± 5 s /day	Max. ± 5 s /day
Digital inputs		
Number	6	12
Electrical isolation	No	No
Input voltage L1 • Signal 0 • Signal 1	 < 40 V AC > 79 V AC	 < 40 V AC > 79 V AC
Input current at • Signal 0 • Signal 1	 < 0.03 mA > 0.08 mA	 < 0.03 mA > 0.08 mA

	LOGO! 230RC LOGO! 230RCo	LOGO! 230RCL LOGO! 230RCLB11
Delay time at • 0 after 1 • 1 after 0	 Typically 50 ms Typically 50 ms	 Typically 50 ms Typically 50 ms
Line length (unshielded)	100 m	100 m
Digital outputs		
Number	4	8
Output type	Relay outputs	Relay outputs
Electrical isolation	Yes	Yes
In groups of	1	2
Activation of digital input	Yes	Yes
Continuous current $I_{th \ (per \ connector)}$	Max. 10 A	Max. 10 A
Incandescent lamp load (25,000 switching cycles) at 230/240 V AC 115/120 V AC	 1000 W 500 W	 1000 W 500 W
Fluorescent tubes with electr. control gear (25,000 switching cycles)	10 x 58 W (at 230/240 V AC)	10 x 58 W (at 230/240 V AC)
Fluorescent tubes, conventionally compensated (25,000 switching cycles)	1 x 58 W (at 230/240 V AC)	1 x 58 W (at 230/240 V AC)
Fluorescent tubes, uncompensated (25,000 switching cycles)	10 x 58 W (at 230/240 V AC)	10 x 58 W (at 230/240 V AC)
Short circuit-proof cos 1	Power protection B16 600A	Power protection B16 600A
Short-circuit proof cos 0.5 to 0.7	Power protection B16 900A	Power protection B16 900A

	LOGO! 230RC LOGO! 230RCo	LOGO! 230RCL LOGO! 230RCLB11
Parallel switching of outputs to increase power	Not permitted	Not permitted
Protection of output relay (if desired)	Max. 16 A, characteristic B16	Max. 16 A, characteristic B16
Switching rate		
Mechanical	10 Hz	10 Hz
Ohmic load/lamp load	2 Hz	2 Hz
Inductive load	0.5 Hz	0.5 Hz
ASi-slave connection (LOGO! 230RCLB11 only)		
ASi profile • I/O config • ID code		7.F 7_h F_h
Number of virtual digital inputs		4
Number of virtual digital outputs		4
Power supply		ASi power supply unit
Power consumption		Typically 30 mA
Electrical isolation		Yes
Reverse polarity protection		Yes

A.3 Technical specifications: LOGO! 24, ...24RC, ...24RCo

	LOGO! 24	LOGO! 24RC LOGO! 24RCo
Power supply		
Input voltage	24 V DC	24 V AC/DC
Permissible Range	20.4 to 28.8 V DC	20.4 ... 26.4 V AC 20.4 to 28.8 V DC
Power consumption from 24 V DC	10 ... 20 mA	10 ... 65 mA (DC) 15 ... 75 mA (AC)
Voltage failure bridging		Typically 5 ms
Power loss at 24 V	0.2 ... 0.5 W	0.2 ... 1.6 W (DC) 0.3 ... 1.8 W (AC)
Clock buffering at 25 °C		Typically 80 h
Accuracy of the real-time clock		Max. ± 5 s /day
Digital inputs		
Number	6	6
Electrical isolation	No	No
Input voltage L+ • Signal 0 • Signal 1	< 5 V DC > 12 V DC	< 5 V AC/DC > 12 V AC/DC
Input current at • Signal 0 • Signal 1	< 0.3 mA > 1.5 mA	< 1.0 mA > 2.5 mA
Delay time at • 0 to 1 • 1 to 0	Typically 1.5 ms Typically 1.5 ms	Typically 1.5 ms Typically 15 ms
Line length (unshielded)	100 m	100 m

	LOGO! 24	**LOGO! 24RC** **LOGO! 24RCo**
Digital outputs		
Number	4	4
Output type	Transistor, current-sourcing	Relay outputs
Electrical isolation	No	Yes
In groups of		1
Activation of digital input	Yes	
Output voltage	≜ Supply voltage	
Output current	Max. 0.3 A	
Continuous current I_{th}		Max. 10 A
Incandescent lamp load (25,000 switching cycles)		1000 W
Fluorescent tubes with electr. control gear (25,000 switching cycles)		10 x 58 W
Fluorescent tubes, conventionally compensated (25,000 switching cycles)		1 x 58 W
Fluorescent tubes, uncompensated (25,000 switching cycles)		10 x 58 W
Short circuit-proof and overload-proof	Yes	
Short-circuit current limiting	Approx. 1 A	
Derating	None throughout the entire temperature range	
Short circuit-proof cos 1		Power protection B16 600A

	LOGO! 24	**LOGO! 24RC** **LOGO! 24RCo**
Short-circuit proof cos 0.5 to 0.7		Power protection B16 900A
Parallel switching of outputs to increase power	Not permitted	Not permitted
Protection of output relay (if desired)		Max. 16 A, characteristic B16
Switching rate		
Mechanical		10 Hz
Electrical	10 Hz	
Ohmic load/lamp load	10 Hz	2 Hz
Inductive load	0.5 Hz	0.5 Hz

A.4 Technical specifications: LOGO! 24L, ...24RCL, ...24RCLB11

	LOGO! 24L	LOGO! 24RCL LOGO! 24RCLB11
Power supply		
Input voltage	24 V DC	24 V DC
Permissible range	20.4 ... 28.8 V DC	20.4 to 28.8 V DC
Power consumption from 24 V DC	10 ... 30 mA + 0.3 A per output	15 ... 120 mA
Voltage failure bridging		Typically 5 ms
Power loss at 24 V DC	0.2 ... 0.8 W	0.3 ... 2.9 W
Clock buffering at 25		Typically 80 h
Accuracy of the real-time clock		Max. ± 5 s /day
Electrical isolation	No	No
Reverse polarity protection	Yes	Yes
Digital inputs		
Number	12	12
Electrical isolation	No	No
Input voltage L+ • Signal 0 • Signal 1	< 5 V DC > 12 V DC	< 5 V DC > 12 V DC
Input current at • Signal 0 • Signal 1	< 1.5 mA > 4.5 mA	< 1.5 mA > 4.5 mA
Delay time at • 0 to 1 • 1 to 0	Typically 1.5 ms Typically 1.5 ms	Typically 1.5 ms Typically 1.5 ms

	LOGO! 24L	LOGO! 24RCL LOGO! 24RCLB11
Line length (unshielded)	100 m	100 m
Digital outputs		
Number	8	8
Output type	Transistor,current-sourcing	Relay outputs
Electrical isolation	No	Yes
In groups of		2
Activation of digital input	Yes	Yes
Output voltage	⚌ Supply voltage	
Output current	Max. 0.3 A	
Continuous current I_{th} (per connector)		Max. 10 A
Incandescent lamp load (25,000 switching cycles)		1000 W
Fluorescent tubes with electr. control gear (25,000 switching cycles)		10 x 58 W
Fluorescent tubes, conventionally compensated (25,000 switching cycles)		1 x 58 W
Fluorescent tubes, uncompensated (25,000 switching cycles)		10 x 58 W
Short circuit-proof and overload-proof	Yes	
Short-circuit current limiting	Approx. 1 A	
Derating	None throughout the entire temperature range	None throughout the entire temperature range
Short circuit-proof cos 1		Power protection B16 600A

Technical specifications

	LOGO! 24L	LOGO! 24RCL LOGO! 24RCLB11
Short-circuit proof cos 0.5 to 0.7		Power protection B16 900A
Parallel switching of outputs to increase power	Not permitted	Not permitted
Protection of output relay (if desired)		Max. 16 A, characteristic B16
Switching rate		
Mechanical		10 Hz
Electrical	10 Hz	
Ohmic load/lamp load	10 Hz	2 Hz
Inductive load	0.5 Hz	0.5 Hz
ASi slave connection(LOGO! 24RLCB11 only)		
ASi profile • I/O config • ID code		7.F 7_h F_h
Number of virtual digital inputs		4
Number of virtual digital outputs		4
Power supply		ASi power supply unit
Power consumption		Typically 30 mA
Electrical isolation		Yes
Reverse polarity protection		Yes

A.5 Technical specifications: LOGO! 12...

	LOGO! 12RC	LOGO! 12RCL
Power supply		
Input voltage	12 V DC	12 V DC
Permissible range	10.8 ... 15.6 V DC	10.8 ... 15.6 V DC
Power consumption from 12 V DC	10 ... 100 mA	10 ... 165 mA
Voltage failure bridging	Typically 2 ms	Typically 5 ms
Power loss at 12 V DC	0.1 ... 1.2 W	0.1 ... 2.0 W
Clock buffering at 25 ˚C	Typically 80 h	Typically 80 h
Accuracy of the real-time clock	Max. ± 5 s /day	Max. ± 5 s /day
Electrical isolation	No	No
Reverse polarity protection	Yes	Yes
Digital inputs		
Number	6	12
Electrical isolation	No	No
Input voltage L+		
• Signal 0	< 4 V DC	< 4 V DC
• Signal 1	> 8 V DC	> 8 V DC
Input current at		
• Signal 0	< 0.5 mA	< 0.5 mA
• Signal 1	> 1.5 mA	> 1.5 mA
Delay time at		
• 0 to 1	Typically 1.5 ms	Typically 1.5 ms
• 1 to 0	Typically 1.5 ms	Typically 1.5 ms
Line length (unshielded)	100 m	100 m
Digital outputs		
Number	4	8

	LOGO! 12RC	LOGO! 12RCL
Output type	Relay outputs	Relay outputs
Electrical isolation	Yes	Yes
In groups of	1	2
Activation of digital input	Yes	Yes
Output voltage		
Output current		
Continuous current $I_{th\ (per\ connector)}$	Max. 10 A	Max. 10 A
Incandescent lamp load (25,000 switching cycles) at	1000 W	1000 W
Fluorescent tubes with electr. control gear (25,000 switching cycles)	10 x 58 W	10 x 58 W
Fluorescent tubes, conventionally compensated (25,000 switching cycles)	1 x 58 W	1 x 58 W
Fluorescent tubes, uncompensated (25,000 switching cycles)	10 x 58 W	10 x 58 W
Short circuit-proof and overload-proof		
Short-circuit current limiting		
Derating	None throughout the entire temperature range	None throughout the entire temperature range
Short circuit-proof cos 1	Power protection B16 600A	Power protection B16 600A
Short-circuit proof cos 0.5 to 0.7	Power protection B16 900A	Power protection B16 900A
Parallel switching of outputs to increase power	Not permitted	Not permitted

	LOGO! 12RC	**LOGO! 12RCL**
Protection of output relay (if desired)	Max. 16 A, characteristic B16	Max. 16 A, characteristic B16
Switching rate		
Mechanical	10 Hz	10 Hz
Electrical		
Ohmic load/lamp load	2 Hz	2 Hz
Inductive load	0.5 Hz	0.5 Hz

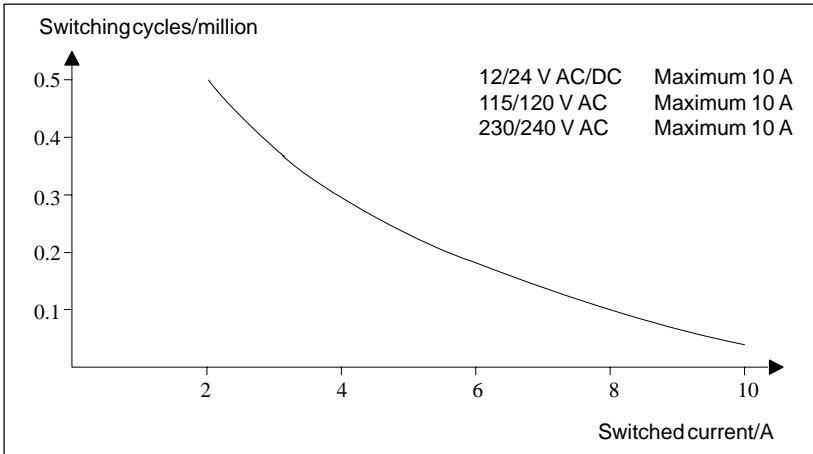**Switching capacity and service life of the relay outputs**

Ohmic load

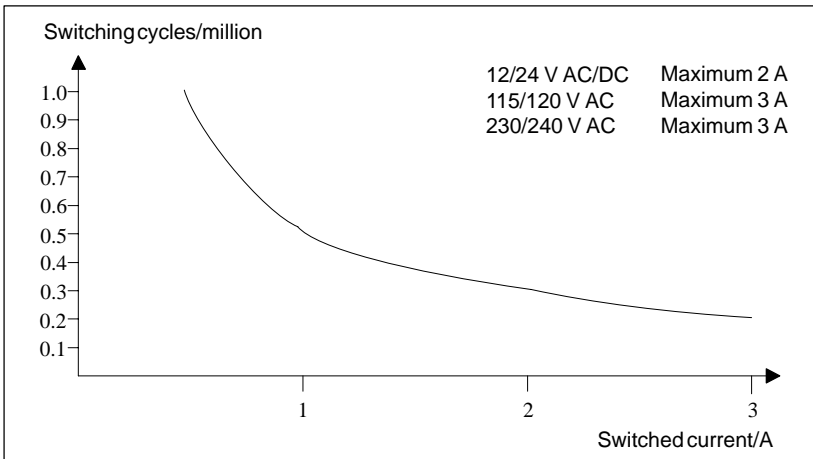Figure A   Switching capacity and service life of the contacts at ohmic load (heating)

Inductive load

Figure B   Switching capacity and service life of the contacts at highly inductive load to IEC 947-5-1 DC13/AC15 (contactors, solenoid coils, motors)

A.6 Technical specifications: LOGO!Power 12 V

LOGO! Power 12 V is a switched-mode primary power supply unit for LOGO! devices. Two current intensities are available.

	LOGO! Power 12 V / 1.9 A	LOGO! Power 12 V / 4.5 A
Input data		
Input voltage	120 ... 230 V AC	
Permissible range	85 ... 264 V AC	
Permissible mains frequency	47 ... 63 Hz	
Voltage failure bridging	> 40 ms (at 187 V AC)	
Input current	0.3 ... 0.18 A	0.73 ... 0.43 A
Making current (25˚C)	≤ 15 A	≤ 30 A
Device protection	Internal	
Recommended circuit breaker (IEC 898) in mains in-feed	> 6 A characteristic D > 10 A characteristic C	
Output data		

Output voltage
   Overall tolerance
   Adjustment range
   Residual ripple
	12 V DC
	+/- 3 %
	11.1 ... 12.9 V DC
	< 200 mV$_{SS}$	

Output current Overcurrent limiting	1.9 A 2.4 A	4.5 A 4.5 A
Efficiency	≥ 80 %	
Parallel switching to increase power	Yes	
Electromagnetic compatibility		
Interference suppression	EN 50081-1, EN 55022 Class B	
Interference immunity	EN 50082-2	

	LOGO! Power 12 V / 1.9 A	LOGO! Power 12 V / 4.5 A
Safety		
Electrical isolation, primary/ secondary	Yes, SELV (to EN 60950/VDE 0805)	
Safety class	II (to IEC 536/VDE 0106 T1)	
Type of protection	IP 20 (to EN 60529/VDE 470 T1)	
CE marking UL/CSA certification FM approval	Yes Yes UL 508 / CSA 22.2 In preparation	
General details		
Ambient temperature range	–20 ... +55˚C, natural convection	
Storage and transport temperature	- 40 ... +70˚C	
Connections on input	One connector (1x2.5mm^2 o. 2x 1.5 mm^2) each for L1 and N	
Connections on output	Two connectors (1x2.5mm^2 o. 2x 1.5 mm^2) each for L+ and M	
Installation	On 35 mm DIN rail, snap-on	
Dimensions in mm (WxHxD)	72 x 80 x 55	126 x 90 x 55
Approx. weight	0.2 kg	0.4 kg

A.7 Technical specifications: LOGO!Power 24 V

LOGO! Power 24 V is a switched-mode primary power supply unit for LOGO! devices. Two current intensities are available.

	LOGO! Power 24 V/1.3 A	LOGO! Power 24 V/2.5 A
Input data		
Input voltage	120 ... 230 V AC	
Permissible range	85 ... 264 V AC	
Permissible mains frequency	47 ... 63 Hz	
Voltage failure bridging	40 ms (at 187 V AC)	
Input current	0.48 ... 0.3 A	0.85 ... 0.5 A
Making current (25˚C)	< 15 A	< 30 A
Device protection	Internal	
Recommended circuit breaker (IEC 898) in mains in-feed	> 6 A characteristic D > 10 A characteristic C	
Output data		
Output voltage Overall tolerance Adjustment range Residual ripple	24 V DC +/- 3 % 22.2 ... 25.8 V DC < 250 mV$_{SS}$	
Output current Overcurrent limiting	1.3 A 1.6 A	2.5 A 2.8 A
Efficiency	< 80 %	
Parallel switching to increase power	Yes	
Electromagnetic compatibility		
Interference suppression	EN 50081-1, EN 55022 Class B	
Interference immunity	EN 50082-2	

	LOGO! Power 24 V/1.3 A	LOGO! Power 24 V/2.5 A
Safety		
Electrical isolation, primary/ secondary	Yes, SELV (to EN 60950/VDE 0805)	
Safety class	II (to IEC 536/VDE 0106 T1)	
Type of protection	IP 20 (to EN 60529/VDE 470 T1)	
CE marking UL/CSA certification FM approval	Yes Yes UL 508 / CSA 22.2 Yes Class I, Div. 2, T4	
General details		
Ambient temperature range	–20 ... +55˚C, natural convection	
Storage and transport temperature	- 40 ... +70˚C	
Connections on input	One connector (1x2.5mm^2 o. 2x 1.5 mm^2) each for L1 and N	
Connections on output	Two connectors (1x2.5mm^2 o. 2x 1.5 mm^2) each for L+ and M	
Installation	On 35 mm DIN rail, snap-on	
Dimensions in mm (WxHxD)	72 x 80 x 55	126 x 90 x 55
Approx. weight	0.2 kg	0.4 kg

A.8 Technical specifications: LOGO! Contact 24/230

LOGO! Contact 24 and LOGO! Contact 230 are switching modules for the direct switching of ohmic loads of up to 20 A and motors of up to 4 kW (without noise emission, hum-free).

	LOGO! Contact 24	**LOGO! Contact 230**
Operating voltage	24 V DC	230 V AC; 50/60 Hz
Switching capacity		
Utilization category AC-1: Switching of ohmic load at 55˚C	85 ... 264 V (derating at <93 V)	
Operating current at 400 V	20 A	
Output of three-phase loads at 400 V	13 kW	
Utilization category AC-2, AC-3: Motors with slipring or squirrel-cage rotor	85 ... 264 V (derating at <93 V)	
Operating current at 400 V	8.4 A	
Output of three-phase loads at 400 V	4 kW	
Short-circuit protection: Assignment type 1 Assignment type 2	25 A 10 A	
Connecting leads	Finely stranded with wire end ferrules single-core 2x (0.75 to 2.5) mm^2 2x (1 to 2.5) mm^2 1 x 4 mm^2	
Dimensions (WxHxD)	36 x 72 x 55	
Ambient temperature	-25 ... +55˚C	
Storage temperature	-50 ... +80˚C	

B Determining memory require-ments

Use of memory areas

If you are unable to enter another block when you are entering a program, this means a memory area is completely occupied. LOGO! only offers you the blocks that will still fit into LOGO!. If no more blocks from a list fit into LOGO!, you will not be able to select the list again.

If a memory area is occupied, you have to optimize your circuit or use a second LOGO!.

Determining the amount of memory required

When you calculate the memory requirements of a circuit, you must always include all the individual areas of the memory.

Example:

Par	RAM	Timer	REM
6	2	0	0

Par	RAM	Timer	REM
0	0	0	0
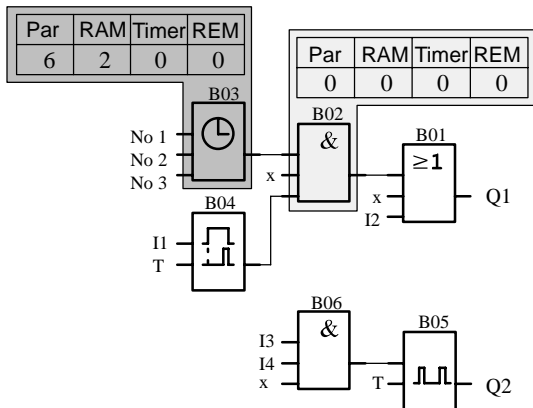
The example program consists of the following:

Block no.	Function	Par	RAM	Timer	REM	Blocks
B01	OR	0	0	0	0	1
B02	AND	0	0	0	0	1
B03	Time switch	6	2	0	0	1
B04	On-delay	1	1	1	0	1
B05	Clock pulse generator	1	1	1	0	1
B06	AND	0	0	0	0	1
	Resources occupied by the program	8	4	2	0	6
	Memory limitations in LOGO!	48	27	16	15	56
	Still available in LOGO!	40	23	14	15	50

The program therefore fits into LOGO!.

C Determining the cycle time

The time required for a program cycle can be determined using a short test program. The test program is created in LOGO! and produces a value during processing in parameterization mode from which the current cycle time can be calculated.

Test program

1. Create the test program by linking an output or marker to a threshold switch and by switching a clock pulse generator at its input, which is switched on by a hi signal.
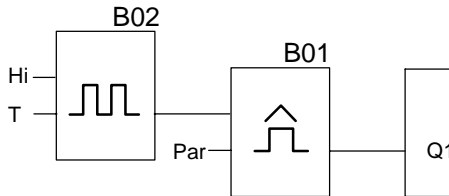
2. Parameterize the two blocks as shown below. A clock pulse is generated in each program cycle based on the cycle time of 0 seconds. The time interval of the threshold switch is set to 2 seconds.

```
B02:T
T =00.00s+
```

```
B01:Par
SW↑=1000+
SW↓=0000
G_T=02.00s
```

3. Then start the program and switch LOGO! to parame-
   terization mode. You can look at the parameters for the
   threshold switch in parameterization mode.

```
B01:Par
SW↑=1000+
SW↓=0000
fa =0086
```

fa= is the sum of the
pulses measured per
time unit G_T

4. The inverse value of fa is equal to the cycle time of
   LOGO! with the current program stored in the memory.

   1/fa = cycle time in s

Explanation

The clock pulse generator (T=0) changes its output signal
every time the program is executed. A level (high or low)
thus lasts exactly one cycle. A period therefore lasts 2
cycles.

The threshold switch indicates the ratio of periods per 2
seconds which results in the ratio of cycles per second.

Edge change of clock pulse generator
each time the program is executed

Cycle time

Periods

1 period = 1 pulse = 2 cycles
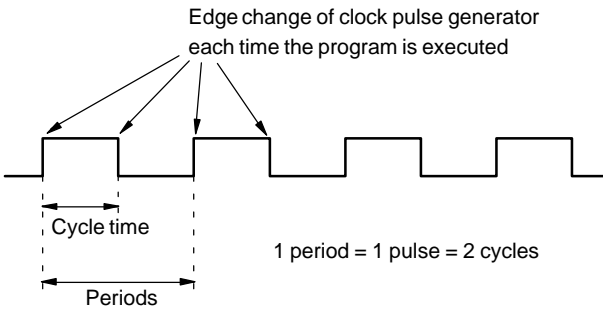
D LOGO! without display

The LOGO! 24RCo and LOGO! 230RCo variants without a display were developed because some special applications don't need operating units such as keypads and a display during operation.
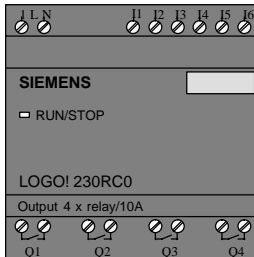
Less is definitely more!

The advantages for you:

- More cost-efficient than with an operating unit.
- Require less space in the cabinet than conventional hardware.
- Considerably more flexible and less costly than separate hardware units.
- Advantageous for applications in which two or three conventional switching devices can be replaced.
- Very easy to use.
- Cannot be used by unauthorized persons.
- Compatible with LOGO! basic variants.

Programming without an operating unit

There are two ways to program LOGO! without a display:

- Create a program with LOGO! software on the PC and transfer the program to LOGO!

- Transfer the program from a LOGO! program module/ card to your LOGO! without a display.

Operating characteristics

Once the power supply has been connected, LOGO! is ready for operation. You can switch off LOGO! without a display by disconnecting the power supply, for example by removing the plug.

You can't use a key combination to set up LOGO! RCo variants for data transfer, and similarly programs can't be stopped or started using keys. LOGO! RCo variants therefore have modified startup characteristics:

Startup characteristics

If a LOGO! program module/card is inserted, a program stored there will be copied to the device immediately after LOGO! has been switched on, thus overwriting an existing program.

If a PC cable is inserted, LOGO! automatically goes into PC ↔ LOGO mode when it is switched on. Using the PC software LOGO!Soft and LOGO!Soft Comfort you can read the programs from LOGO! or store them on LOGO!.

If there is already a valid program in the program memory, LOGO! will automatically transfer from STOP to RUN after power has been switched on.

Operating status indicator

Operating statuses, such as Power On, RUN and STOP are indicated by an LED on the front hood.

- Red LED: PowerOn/STOP
- Green LED: PowerOn/RUN

After the power supply has been switched on and whenever LOGO! is not in RUN mode, the red LED comes on. When LOGO! is in RUN mode, the green LED comes on.

E LOGO! ...LB11:
Switchover active - passive
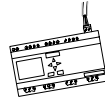
All LOGO!...B11 variants are factory-set to address 0.

When the master is assigning addresses, only one active slave may be set to address 0 on the ASi bus at any one time. All the other slaves with address 0 must be passive i.e. unknown on the bus.

! Caution

The ASi address can be changed 10 times for each LOGO! ...LB11 variant.

Additional changes are not guaranteed.

To enable you to switch LOGO!...B11 to passive, we have integrated a menu item in the programming menu.
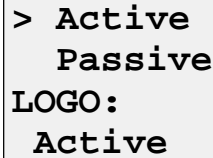
Switching LOGO! ...B11 between active and passive

1. Switch LOGO!...B11 to programming mode (using the 3-finger grip) and then press **OK** to get directly into the programming menu.

2. Press the key three times ▼

   The cursor (>) is now positioned at the start of the ASi_BUS.. line

```
 Edit Prg
 Clear Prg
 Set Clock
>ASi_BUS..
```

3. Press **OK**. The following display appears:

```
> Active
  Passive
LOGO:
 Active
```

4. Switch LOGO!...B11 to passive by pressing the key ▼ and then **OK**. The display indicates the new state

```
LOGO:
 Passive
```

5. As soon as the master has recognized an active slave and assigned it an address, you can switch another slave back from **passive** to **active**.

Note

You can only exit the menu for switching between active and passive if LOGO! is switched to **active**.

Order numbers

Table A

Variant	Designation	Order number
Standard	LOGO! 12RC LOGO! 24 LOGO! 24RC LOGO! 230RC	6ED1 052–1BB00–0BA1 6ED1 052–1CA00–0BA1 6ED1 052–1HB00–0BA1 6ED1 052–1FB00–0BA1
Without display	LOGO! 24RCo LOGO! 230RCo	6ED1 052–2HB00–0BA1 6ED1 052–2FB00–0BA1
Long	LOGO! 12RCL LOGO! 24L LOGO! 24RCL LOGO! 230RCL	6ED1 053–1BB00–0BA1 6ED1 053–1CA00–0BA1 6ED1 053–1HB00–0BA1 6ED1 053–1FB00–0BA1
Bus	LOGO! 24RCLB11 LOGO! 230RCLB11	6ED1 053–1HH00–0BA1 6ED1 053–1FH00–0BA1

Table B

Accessories	Designation	Order number
Software	LOGO!Soft	6ED1 058–1AA00–0BA1
	LOGO!Soft Comfort	6ED1 058–1BA00–0YC0
Program modules	Yellow card	6ED1 056–1BA00–0AA0
	Red card	6ED1 056–4BA00–0AA0
Switching module	LOGO!Contact 24 V	6ED1 057–4CA00–0AA0
	LOGO!Contact 230 V	6ED1 057–4EA00–0AA0
Power modules	LOGO!Power 12V/1.9A	6EP1 053–321–1SH01
	LOGO!Power 12V/4.5A	6EP1 322–1SH01
	LOGO!Power 24V/1.3A	6EP1 331–1SH01
	LOGO!Power 24V/2.5A	6EP1 332–1SH41
Others	PC cable	6ED1 057–1AA00–0BA0
	Manual	6ED1 050–1AA00–0AE2

Abbreviations

B01   Block number B01

B11   LOGO! device designation: ASi bus connection

BN   Block number

C   LOGO! device designation: integrated clock

Cnt   Count = input for counter

Co   Connector

Dir   Direction (e.g. for counter)

En   Enable (e.g. for switching on the clock pulse generator)

BF   Basic functions

L   LOGO! device designation: long variant

No   Cams (time switch parameters)

o   LOGO! device designation: without display

Par   Parameter

R   Reset

R   LOGO! device designation: relay outputs

S   Set (e.g. setting the latching relay)

SF   Special functions

T   Time (parameter)

Trg   Trigger (parameter)

Index

Symbols

?, on the display, 53

Numbers

3-finger grip, 32

4 golden rules, 32

A

Active, 186

Active-passive, switchover, 186

AND, 59

Application
  automatic door, 121
  centralized activation and sur-
    veillance of several indu-
    strial gates, 121
  fluorescent lamps, 121
  industrial gate, 121
  rainwater pump, 121
  stairway, hall or corridor ligh-
    ting, 121
  ventilation system, 121

Applications, 121

AS interface, 3

ASi bus, 17

ASi master, 17

ASi system, 17

ASi_BUS, 186

Asynchronous pulse generator,
  pulse generator, 92

B

B11, 4

Basic functions, 58

BF, 55, 58

Block, 22
  deleting, 51
    a number of interconnected
      blocks, 52

Block number, assigning a , 27

Block numbers, 26

Blocks, 26

BN, 55

Bus connector cable, 18

C

CE marking, 6

Certification, 6

Changing, 45

Checking , 49

Circuit diagram, 29
  from... to LOGO!, 29

Circuit state change, 12

Clock, examples, 75