

ROBOTICS LT Beginner Set	p. 28
Control	p. 30
USB power supply "USB powered"	p. 31
ROBO Pro Light	p. 31
Installation ROBO Pro Light and USB driver	p. 31
Starting the ROBO Pro Light program	p. 32
Merry-go-round – entry level programing	p. 33
Pedestrian light	p. 41
Lighthouse with blinking light	p. 42
Refrigerator	p. 43
Washing machine	p. 44
Sliding door	p. 46
Staircase lighting	p. 47
Windshield wiper	p. 48
Hand dryer	p. 50
Barrier	p. 50
Motor test bench	p. 51
Machine	p. 51
If something doesn't work right ...	p. 52

Contents



ROBOTICS LT Beginner Set



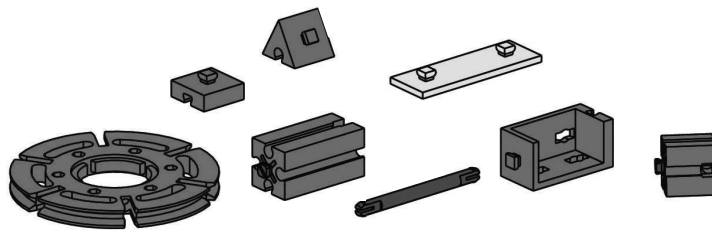
"Hi, first I'd like to introduce myself. I am your fischertechnik trainer and will accompany you through the entire book. I will often give you valuable tips and information."

Components

"First, I would like to introduce you to the most important components, which are contained in the ROBOTICS LT Beginner Set, and give you some technical information about these."

Building blocks

The various building blocks are used to build the individual models. The picture shows a small selection of these.



Electrotechnical components

They look similar to the normal building blocks and can also be put together with these. They work with electrical power. Depending on the function, you call them actuators or sensors.

Actuators

They are called actuators because they are active, they do something. Components such as a motor or an indicator light perform certain work as soon as they are supplied with electrical power. The motor rotates and the indicator light is illuminated.

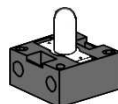
Bulb

The light bulb is an artificial light source. In it, an electrical conductor is heated by electrical power and this causes it to glow.

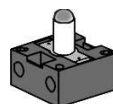
The construction set contains two different lights, the bulb lamp and the lens tip lamp. In the bulb lamp, the light, which is emitted, is dispersed. The light from the lens tip lamp is focused to the front by a lens. This light is used, for example, for light barriers.



Circuit symbol



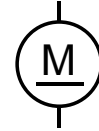
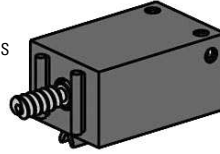
Bulb lamp



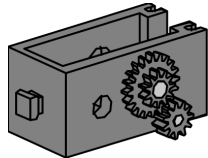
Lens tip lamp

XS Motor

The direct current motor converts electrical energy into mechanical energy. This results in the rotational movement of the motor.



Circuit symbol



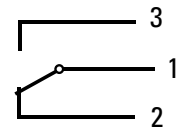
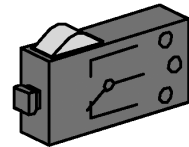
A motor also has a gearbox. You can reduce the speed of the motor with this gearbox.

Sensors

Sensors, such as push-button switches or phototransistors, are components, with which actuators can be controlled. For example, a motor can be turned on or turned off with a push-button switch.

Push-button switches

Push-button switches are in the category of touch sensors. If you push the red button, then a contact is mechanically moved in the casing and electric current flows between the connections 1 and 3. At the same time, the contact between the connections 1 and 2 is interrupted.

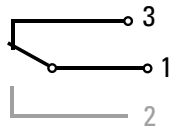


Circuit symbol

So, you can use the push-button switch in two different ways:

As a "closer":

Contacts 1 and 3 are connected. Push-button switch is pressed: electricity flows.

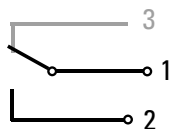


Push-button switch is not pressed: no electricity flows.

The push-button switch shown here is pressed.

As a "break contact":

Contacts 1 and 2 are connected. Push-button switch is pressed: no electricity flows.

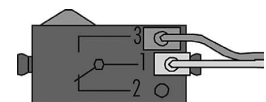


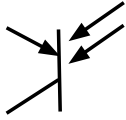
Push-button switch is not pressed: electricity flows.

The push-button switch shown here is pressed.



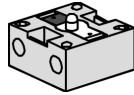
In the construction set, ROBOTICS LT Beginner Set, the push-button switch is always connected to the contacts 1 and 3.





Circuit symbol

Phototransistor



A phototransistor is an electronic switch, which reacts to light. Of course, you have asked yourself how the entrance door in the department store opens automatically without you having to push a push-button switch or use a switch.

A light barrier is used to do this and this barrier consists of a light source (transmitter) and a sensor (receiver). In the construction set, a lens tip lamp is used as the transmitter and a phototransistor as the receiver.

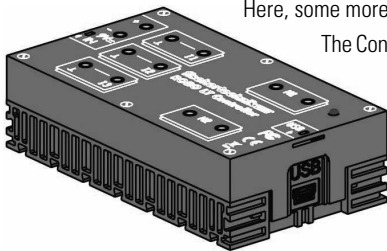
"So, now you have become familiar with the individual components. Perhaps you learned about these in school so that these are nothing new for you."



Control ROBO LT Controller

■ The ROBO LT Controller is the heart of this ROBOTICS construction set. Although you cannot see them, the Controller module has a number of electronic components. The interplay of these components with your computer and a control program allows you to control the models in the ROBOTICS LT Beginner Set.

Here, some more technical information:



The Controller has three inputs (I1-I3) for connection of sensors (mini-switches, phototransistors), two outputs (M1-M2) for connection of actuators (motors, lights), an LED as operation indicator, a DC jack for connecting a power supply unit (9 VDC), two connecting sockets for a 9V battery tray or rechargeable battery pack, and a mini-USB connection for data transmission from the PC to the Controller.

■ The ROBO LT Controller is supplied with power from a PC over a miniature USB connector. Connection of an additional 9V power supply (power pack or battery pack) is not required for the models in the ROBOTICS LT Beginner Set.

An additional 9V power supply is required only for models requiring more current than the USB interface can provide (above approx. 300 mA, e.g. models with several large motors). This is not the case for the models in the LT Beginner Set. For this reason, the construction set does not contain an additional power supply.

■ ROBO Pro Light is the computer software, which together with the ROBO LT Controller you use to control your models. This is found on the CD ROM, which is included in the construction set. First, you must install this software on your PC.

■ The installation of the software, ROBO Pro Light, and the USB driver for the ROBO LT Controller is described in separate installation instructions for this, which are contained in the construction set. The PC needs the USB driver to be able to exchange data with the ROBO LT Controller.

Caution!

You need administrator rights on the PC for the installation of USB drivers. The best thing to do for this step is to let an experienced PC user help you.

USB

power supply

“USB powered”

ROBO Pro Light

Installation

ROBO Pro Light

and USB driver

Starting the ROBO Pro Light program



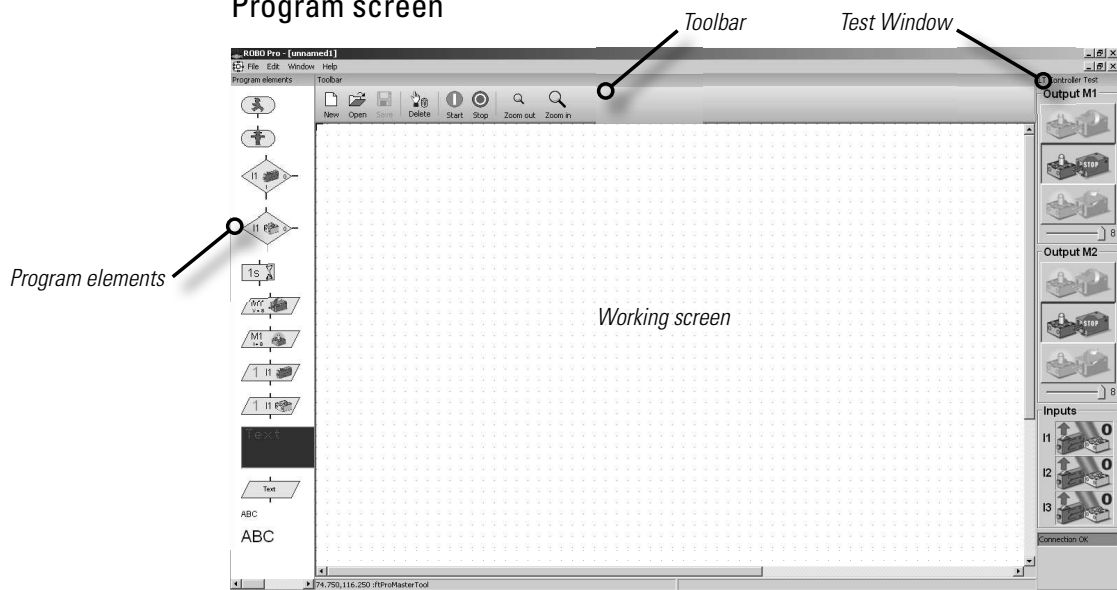
"I am certain that you can do this without my help. Is exactly the same as when you open a game or want to work with the Internet. But here is a short explanation in spite of this."

You find the program, ROBO Pro Light, in the Windows start window at:



These symbols may look somewhat different depending on the Windows version.

Program screen



On the left side, you can find all program elements, which you need for the creation of a control program. The top line is called the toolbar and you have certainly seen this in other programs. Here, there are, for example, menus for Save, Open or the Start of a program. The big window is the working screen. Here, you create the control program. The inputs and outputs for the LT Controller are shown on the right in the test window. Here, you can turn the actuators on and off with the mouse, for example, to make a test to determine if they are connected correctly. Just connect a motor to M1 on the Controller and then turn it on and off with the mouse. In addition, you can check to see if your sensors are working properly. This shows, for example, if a push-button switch is pressed or is not pressed. In addition, the green bar shows you if the connection between the PC and the LT Controller is working properly.

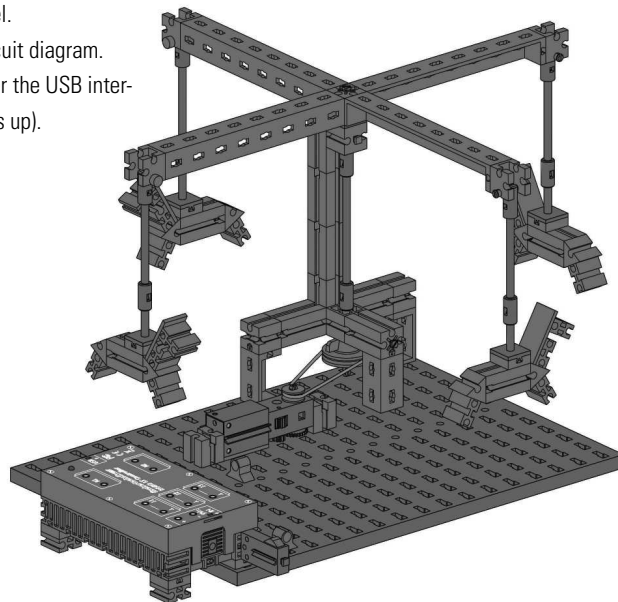
"So, this was some theory at the beginning. Now, you would certainly like to build your first model and control it with the computer as well."



Merry-go-round – entry level programming

■ You can find various models of merry-go-rounds at public festivals and amusement parks. The first motor driven merry-go-round was placed in operation on January 1, 1863 in Bolton in England.

- Using the assembly instructions, build the model.
- Connect the electric cables according to the circuit diagram.
- Connect the ROBO LT Controller with the PC over the USB interface (green LED on the ROBO LT Controller lights up).
- Start the software, ROBO Pro Light.



Task 1

The merry-go-round is to be turned on and off with the mouse using the test window for the software, ROBO Pro Light. Which direction of rotation of the motor (counterclockwise or clockwise) allows the model to rotate correctly?

Press the push-button switch on the model. Now, on the screen, you should see how I1 changes from 0 to 1 when the red button is pushed.



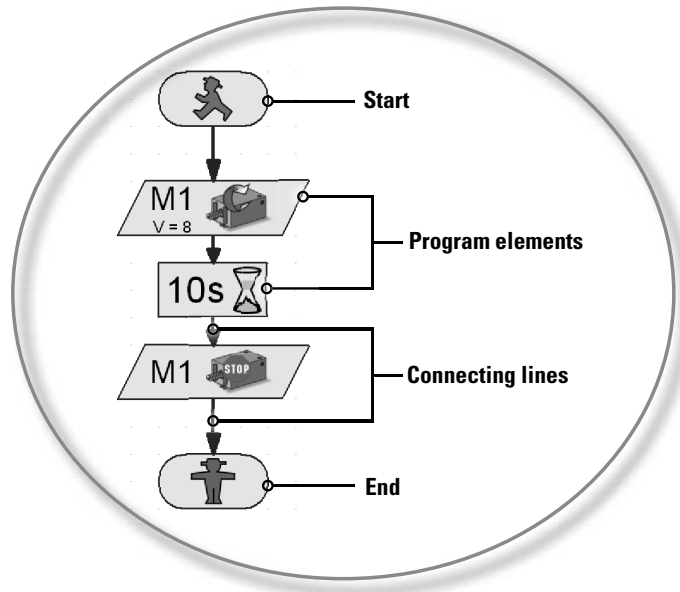
Note!

If the value for the push-button switch does not change from 0 to 1 but from 1 to 0 when it is pressed then you have not connected the plugs to the push-button switch correctly (also see the description push-button switch). One plug is to be connected to connection 1 and the other plug to connection 3.

What is a control program?

A control program tells the merry-go-round what it is to do. The software, ROBO Pro Light, has such a program consisting of various program elements, which are put together with arrows to form a sequence.

"Next, we want to create a control program together."



When the program is started, the elements are carried out one after the other. A control program in ROBO Pro begins with a green little traffic light figure (Start element) and ends with a red little traffic light figure.

Task 2

Can you imagine what the sequence shown causes your merry-go-round to do?



Solution: The merry-go-round motor is started, runs 10 seconds and then it is turned off.

Now it is your turn and you can create your first control program.

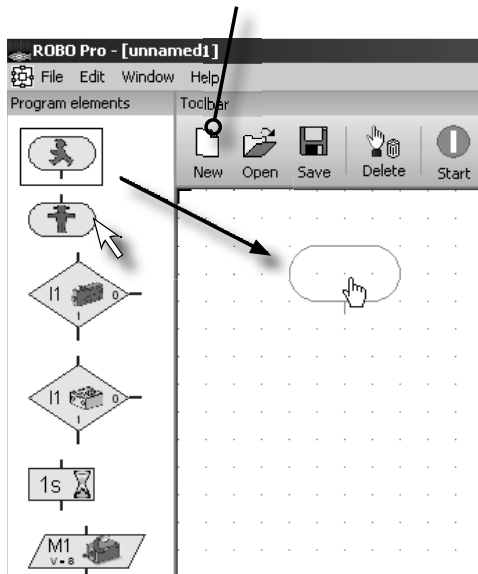
Task 3

The merry-go-round is to be started with the push-button switch (I1). It is to rotate for 10 seconds and then stop.




You proceed as follows:

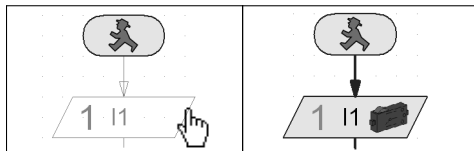
- The ROBO Pro Light software is started, the ROBO LT Controller is connected to the PC and supplied with electricity from the USB interface (green LED on Controller lights up).
- With the button, **File-New**, you begin a new control program.

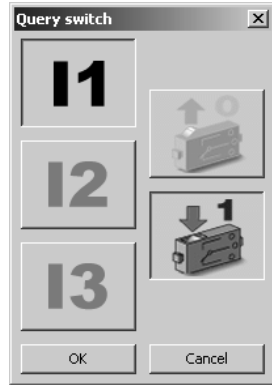


- You put the required program elements on the working screen with the mouse. You start with the green little traffic light figure. Click on the element with the left mouse button and just drag it onto the working screen. By releasing or again clicking the left mouse button, you place the element at the desired position somewhere on the working screen.

- As is described in the task, the merry-go-round meaning the motor M1 is to be started when the push-button switch I1 is pressed. In order to query the switch, you drag the element,  **Query switch**, onto the working screen directly below the Start element. Both elements are automatically connected with each other.

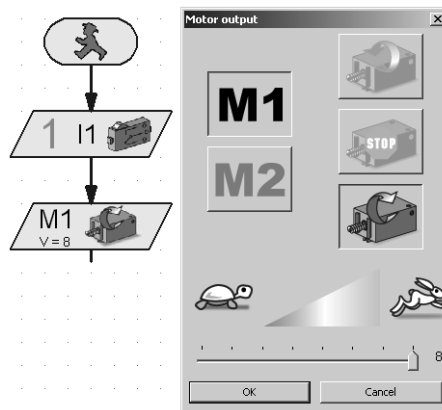
- Move the mouse pointer over the element, which has just been inserted. As soon as the cursor turns into a hand, you can drag the element when the **left mouse** button is pressed. If you press the **right mouse button**, the following dialogue box opens:





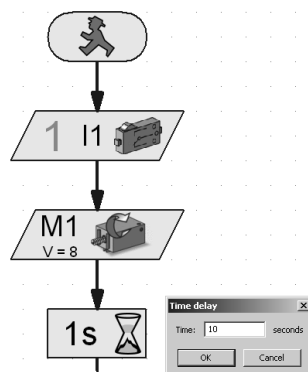
In this window, you can set which push-button switch (I1-I3) you want to query and if the switch is pressed (1) or not pressed (0). I1 and "switch pressed (1)" are already preset so that you can confirm with **OK**.

- Now, as the next step, you are to insert the motor element into the sequence. As before, after the insertion, you open the dialogue box with the right mouse button.



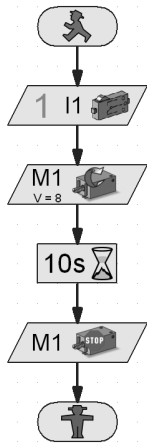
Here, you set the direction of rotation "clockwise" (green arrow). The connection M1 is already preset. Leave the slider control for the speed set on the far right. Then, the motor runs at full speed. Now, you can close the window with **OK**.

- Now, insert the element, Time delay.

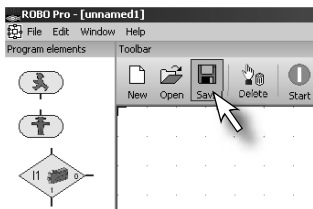


In the dialogue box, set the Time delay to 10 seconds.

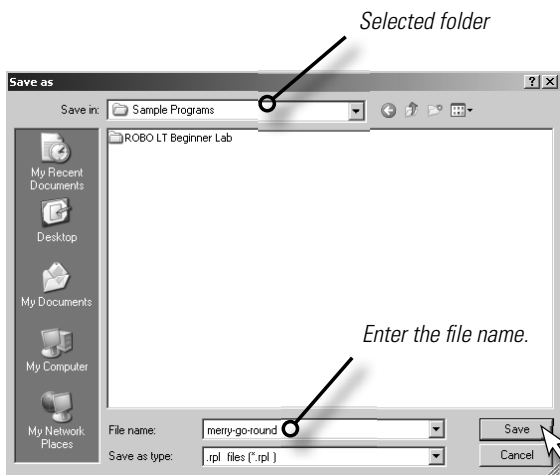
- After this Time delay, the motor is to be turned off. Insert another motor element and in the dialogue box select M1 and Motor stop. Finally, you need the end symbol (red little traffic light figure). Your complete sequence then looks like this:



- Before you try out the program, you should save it so that it doesn't get lost. To do this, go to the



Save button.



A standard Windows window appears. Here, you can select the folder, in which the program is to be saved. In addition, you can give a name to the file, for example, "merry-go-round". The ROBO Pro file automatically receives the extension .rpl. Press the button, **Save**.

Depending on the Windows version, the window may appear somewhat different than the window shown here.



"OK, you are already a big step farther.
Your program is finished and now you
can test it."



Starting and stopping the program

To start the program, click with the mouse on the "Start" button.

As soon as you press the button, the merry-go-round turns for 10 seconds and then stops. The program sequence has arrived at the end symbol.



Beside the button for starting the program, you will find the button for stopping. Regardless of where the program is at the moment, when you press the Stop command, the program is stopped and ended.



"In order to avoid restarting the program after
every program run, you can put a "program
loop" in the program."


Program loop

Task 4

Change the program so that after the motor stops, it does not jump to the end symbol (red little traffic light figure) but returns to Query switch. The end symbol is no longer needed and can be deleted.



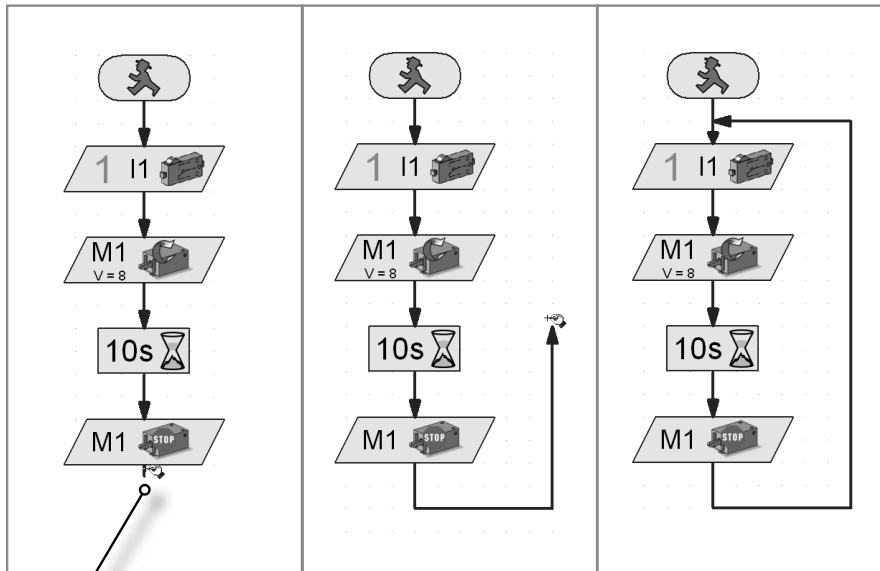
Delete program elements and lines

- Click with the left mouse button on the button "Delete"  and then on the element or the line that you want to delete.
- Another possibility: Move the mouse pointer to the program element or the line that you want to delete and then click with the left mouse button. The element is marked red. Press the "Delete" key on your keyboard. When this is done, the marked element is deleted.

Delete the red little traffic light figure and the line between Motor stop and the little traffic light figure.

Then, draw a connecting line from the end of the element, **Motor stop**, upwards to the connecting line between the start symbol and Query switch.

Draw connecting lines by hand.



End, Start of line


- Move the mouse pointer to the end of the element, Motor stop. The mouse pointer turns into a hand with a pencil.
- Press the left mouse button and then release it. This is the way the line begins. Drag it with the mouse in the desired direction (first downwards).
- If you want to change the direction, then click once with the left mouse button. The line is then bent and you can pull it in another direction (first to the right and then upwards).
- When you get to the connecting line between Start and Query switch then click once again with the left mouse button. This then ends the line. The program loop is finished.

Note!

If you have drawn a line incorrectly by accident and you want to end it in the middle of the action then you can do this by double clicking with the left mouse button and then delete the line.

Save the program, for example, under the name merry-go-round-2 and then try it out. Does it work like you wanted?

Note!

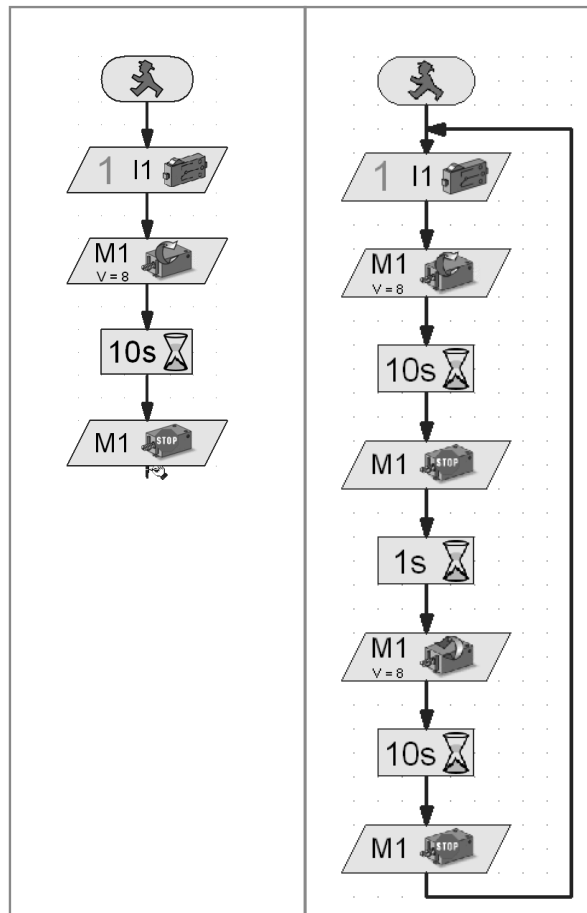
Since the red little traffic light figure is missing, the program must be ended with  the Stop button.

Task 5


Just going in one direction is rather boring. Change the program so that it waits a second after the motor stops and then the merry-go-round turns in the other direction for 10 seconds.



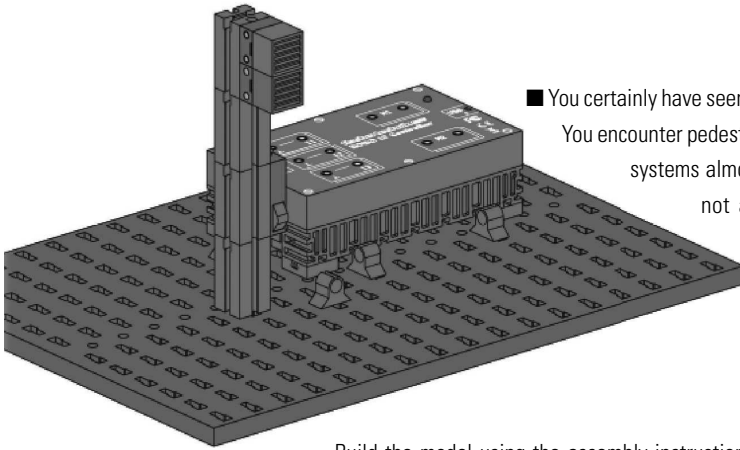
Do you have an idea which program elements you need in addition to solve this task? OK, I'll help you again. When you expand the sequence as shown and then draw a program loop again to the start of the program, the entire thing is taken care of.



Save this program under any name as well, for example, merry-go-round-3, so that you can use it again later.

In order to open an existing program, you press on the button  . A selection window appears, in which you select the folder and the program that you want to open.

So much for the merry-go-round. The construction set contains additional exciting models and programming tasks. Have fun!



■ You certainly have seen various models of traffic lights. You encounter pedestrian lights or entire intersection systems almost daily so that the principle is not anything new for you. Stated simply, lights are turned on and off in a certain sequence.

Build the model using the assembly instructions and then connect the cables according to the circuit diagram.

Pedestrian light

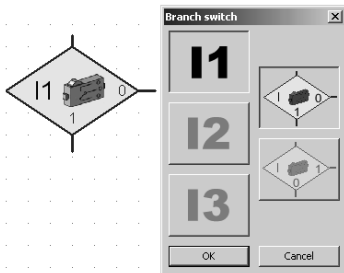
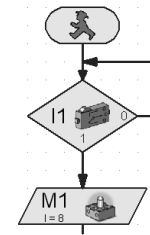


Task 1

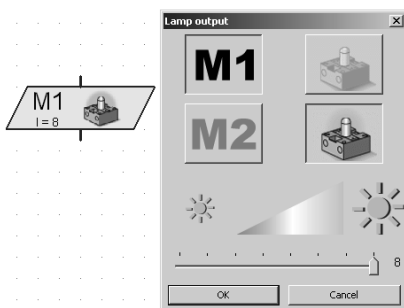
Program a pedestrian light, which is switched with a push-button switch. At first, the light is red. When the push-button switch is pressed, the red phase is to continue for five seconds. This is to be replaced by a green phase, which lasts for 10 seconds. After this, the light returns to red.



Finished program: Pedestrian Light-1.rpl



For querying of the switch, you are to use the command "Branch switch". Here, you have three connections. The program is set as follows: If the push-button switch is not pressed, go to output 0. If the push-button is pressed, go to output 1.



For the command "Lamp output". you also have the possibility of making various settings. You can set the brightness, the output (M1 or M2), and decide if the light is to be turned on or off.

Task 2

As in task 1, program a pedestrian light, which is operated with a push-button switch. After the green phase is ended, the green light is to blink to show that the pedestrian light will change to the red phase in a short time. The green light is to blink three times.

Finished program: Pedestrian Light-2.rpl

Note!

You can find the solution to this task as a finished ROBO Pro Light program at C:\Programs\ROBOPro-Light\Sample Programs\LT-Beginner-Set\Pedestrian Light-2.rpl.

You can also find sample programs in this directory for all other tasks in this activity booklet.

Lighthouse with blinking light



■ Lighthouses stand at important or dangerous points where they serve ships as a navigation mark, which is visible at long distances including at night. With their light signals (beacon), lighthouses show ships the way and thus allow navigation and the bypassing of dangerous positions in the waters.

Build the model using the assembly instructions and connect the cables according to the circuit diagram.

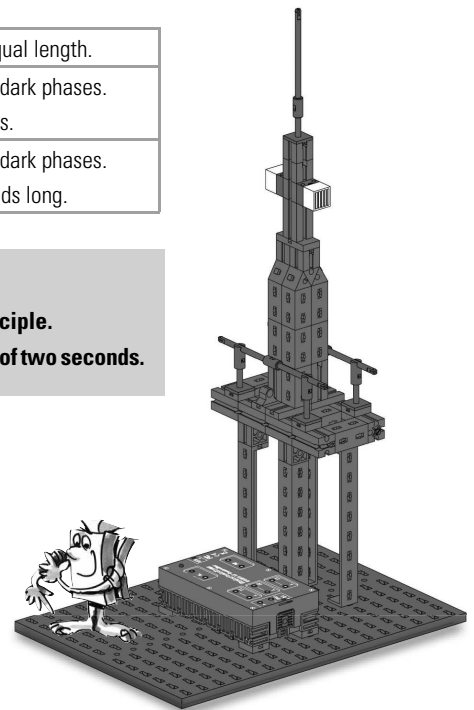
In the nautical charts, the characteristics of a beacon are described by uniform abbreviations, for example:

Intermittent light	The phases of light and dark are of equal length.
Flash	The light phases are shorter than the dark phases. A flash lasts for less than two seconds.
Blinking light	The light phases are shorter than the dark phases. An blinking light is at least two seconds long.

Task 1

Program a "beacon" according to the isophase principle. Set the light and dark phases to be of an equal length of two seconds.

Finished program: Lighthouse-1.rpl



Task 2

Program "lighthouse lighting" according to the flash principle.

For the flash lighting, set the light phase at 0.3 seconds and the dark phase at 1.5 seconds.



Finished program: Lighthouse-2.rpl

Task 3

Program "lighthouse lighting" according to the blink principle.

Both lights are to be illuminated for a different period of time independent of each other.



Finished program: Lighthouse-3.rpl

"Gooood, now let's move to a topic, that you can find at home – a refrigerator. Here, here, I always ask myself if the light is really off, when I close the door?"

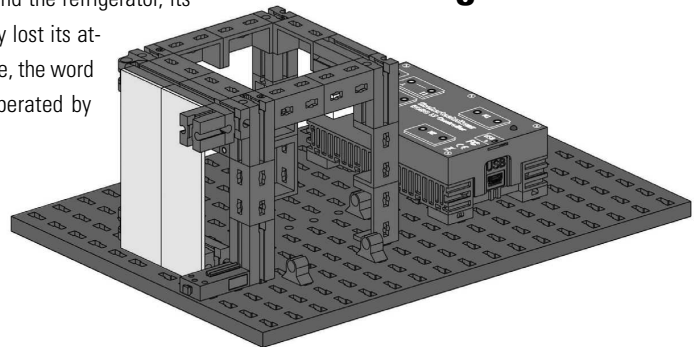


■ With the spread of electrical power and the refrigerator, its forerunner, the ice box without electricity lost its attractiveness. From a historical perspective, the word ice box is still used today but when operated by electricity it is called a refrigerator.

electricity it is called a refrigerator.

Build the model using the assembly instructions and connect the cables according to the circuit diagram.

Refrigerator



Task 1

As soon as the refrigeration door is opened, the white light is to light up. When the door is closed, the light goes out.



Finished program: Refrigerator-1.rpl

Task 2

Supplement task 1: If the refrigerator door is open for more than three seconds then the red indicator light is to blink in addition. It also goes out when the refrigerator door is closed.



Finished program: Refrigerator-2.rpl

Washing machine



■ The most widespread design is the drum washing machine, in which a washing drum rotates around an axis. The advantage of this machine type is the small size so that, for example, it can be installed in kitchen units.

Build the model using the assembly instructions and connect the cables according to the circuit diagram.

And now what all happens during a washing operation? A washing cycle, a spin cycle and a drying cycle are performed.

In the following exercises, you will become familiar with and program various program parts.

"Wash day in the Nineteenth Century:

What a nuisance! Well we have it much easier today with modern machines."

**Task 1**

After you push the Start button, the drum rotates at a low speed for 10 seconds (wash cycle). The display (indicator light on M2) shows the operation of the machine.



Finished program: Washing machine-1.rpl

Task 2

Change the program so that the washing machine only starts when the safety switch for the door is closed.

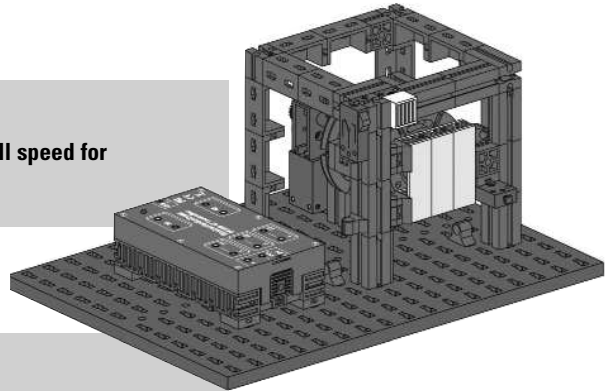


Finished program: Washing machine-2.rpl

Task 3

Place a spin cycle in the program. Here, the motor is to run at full speed for 15 seconds.

Finished program: Washing machine-3.rpl



Task 4

Expand the program with a drying cycle for the laundry. To do this, the drum rotates clockwise slowly at first (10 seconds) then takes a break of three seconds and then rotates counterclockwise for 10 seconds.

Finished program: Washing machine-4.rpl

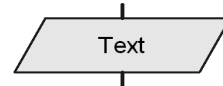


ROBO Pro Light offers the possibility to show certain working cycles such as washing or spinning as text in a display.

For this purpose, use the program element, "Text output".

You can insert this command at any position into your program if you want to send the user some information.

This information is then shown in the display field on the screen. As for other program elements, you can place the display field anywhere on the screen.



Important!

The text, which appears in the display, is overwritten by a new text command. If you don't want to display any text then leave the Text output empty.

Task 5

During the program sequence, the user is always to be shown the current cycle in the text display. When all washing cycles are completed then the end is also to be shown in the display.

Change your program accordingly.

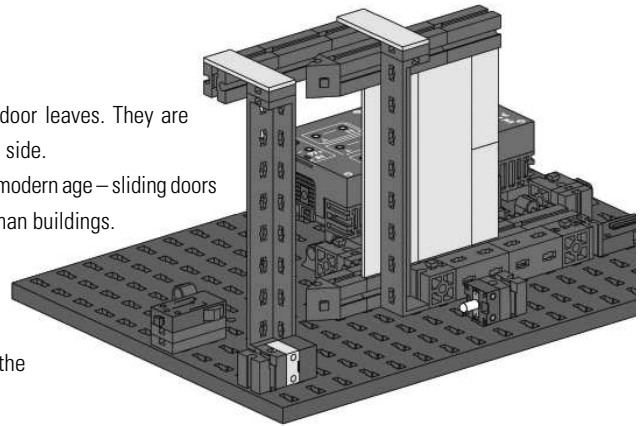
Finished program: Washing machine-5.rpl



Sliding door



■ A sliding door consists of one or more door leaves. They are guided at the top or bottom and open to the side. But don't think that this is an invention of the modern age – sliding doors existed even in the First Century A.D. in Roman buildings. This is shown by excavations in the Italian city of Pompeii. Build the model using the assembly instructions and connect the cables according to the circuit diagram.



Task 1

Create a program, which closes the sliding door, regardless of where it is when the programs starts. Try out various door positions.



Finished program: *Sliding door-1.rpl*

Important!

Push-button switch 1 (I1) is the limit switch for an open door. Push-button switch 2 (I2) is the limit switch for a closed door.

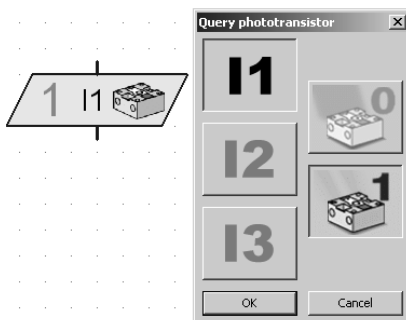
Before you become familiar with the new control command, first look at the associated task.

Task 2

There is a light barrier at the entrance to the sliding door. If the light beam is interrupted, which means that someone wants to enter the business, then the entrance door is opened and it closes automatically after at time of 10 seconds.



Finished program: *Sliding door-2.rpl*



You have already programmed a similar function for the query of the switch in the task for a traffic light.

The program waits at this position until the light beam is interrupted (I3=0). In the dialogue box of the program element, "Query phototransistor", you can decide if the program is to be continued when the light barrier is interrupted (0) or not interrupted (1).

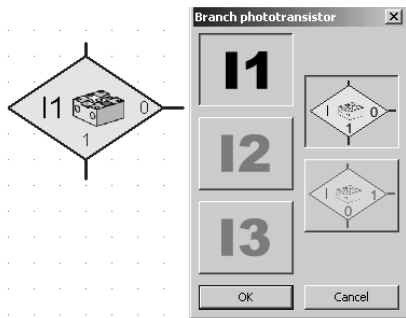
Task 3

So that nobody is caught in the door when it closes, the door is to immediately open when someone interrupts the light barrier. Change the program accordingly. Expand the program with a time delay of five seconds before the door closes.



Finished program: Sliding door-3.rpl

"Ouch"



For querying of the phototransistor, you are to use the command "Branch phototransistor". Here, you have three connections as well. The program is set as follows: If the light barrier is interrupted (no light), go to the output 0. If the light barrier is not interrupted, go to the output 1.

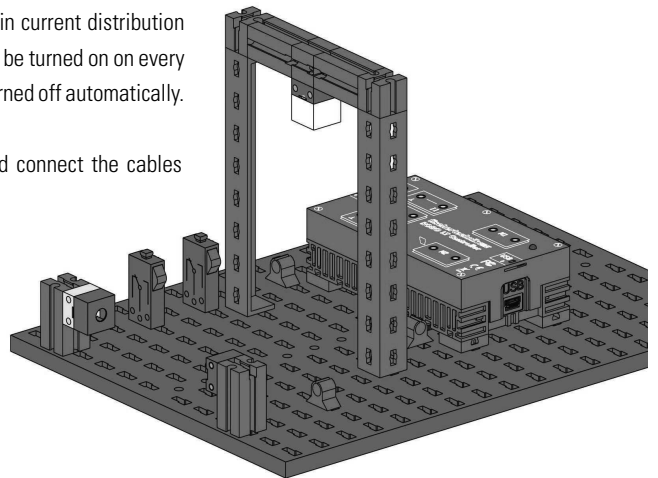
"Do you know about that too?
You turn the light on in the staircase and after a certain time it turns itself off. Or it turns on automatically as soon as you enter the staircase.
How does that work?"



Staircase lighting

■ A small box, which is normally installed in the main current distribution for the building, insures that the staircase lighting can be turned on on every floor. After a time, which can be set, the light is then turned off automatically.

Build the model using the assembly instructions and connect the cables according to the circuit diagram.



Important!

Both push-button switches (I1 and I2) are for the two floors. The motion detector, here a light barrier (I3), is located, for example, in the basement.

Task 1

As soon as one of the two push-button switches is pressed, the staircase lighting is to be turned on. After 10 seconds, it is to go out automatically.



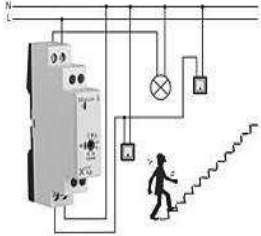
Finished program: Staircase-1.rpl

Task 2

The staircase lighting is to be able to be turned on both with the push-button switches as well as by a light barrier.



Finished program: Staircase-2.rpl

**Task 3**

As soon as one of the two push-button switches is pressed, the staircase lighting is to be turned on. If one of the push-button switches is pressed again, the light goes out.



Finished program: Staircase-3.rpl

Windshield wiper



■ A windshield wiper is a device for the cleaning of the front window or rear window of a vehicle, airplane, ship or a rail vehicle. In November 1903, the American, Mary Anderson, received the patent for the first working windshield wiper system in the world.

Build the model using the assembly instructions and connect the cables according to the circuit diagram.

With the help of two push-button switches, you can query four switch positions and insert them accordingly into your program.

	I1	I2
OFF	Not pressed	Not pressed
Interval	Pressed	Not pressed
Continuous operation slow	Pressed	Pressed
Continuous operation fast	Not pressed	Pressed

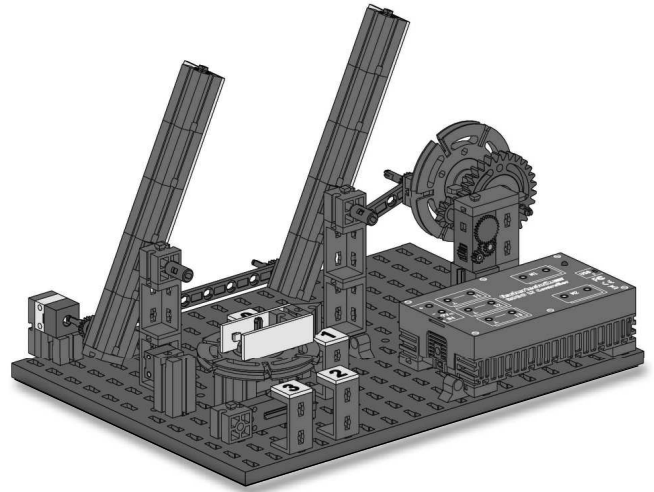
You can use the light barrier to control the interval. For this, the light barrier is interrupted after every wiping cycle.

Task 1

First, a simple task. As soon as the rotary switch in position 1 is moved, the windshield wiper starts to move. If it is reset to 0 then the windshield wiper stops.



Finished program: Windshild wiper-1.rpl



Task 2

If there is heavy rain, the windshield wiper is to move faster. This is to happen with switch position 2. Program your program so that when switched from position 1 to 2, the windshield wiper moves faster and when switched to 1, it is to move with the normal speed again.



Finished program: Windshild wiper-2.rpl



Task 3

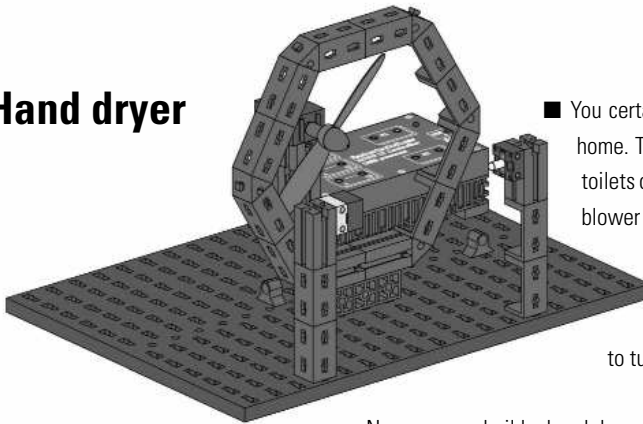
Change the program so that in switch position 1, intermittent switch control is created. In this case, the light barrier is interrupted after every wiping cycle. After two seconds, the wiping cycle is restarted.



Finished program: Windshild wiper-3.rpl



Hand dryer



- You certainly won't have anything like this in your bathroom at home. There you have a towel hanging on a hook. But in public toilets or the restrooms in restaurants there is usually an electric blower mounted on the wall for blowing your hands dry with hot air.

A good invention, particularly when the state-of-the-art appliance doesn't even have a button you have to push to turn it on. Simply hold your hands beneath it to turn it on.

Now you can build a hand dryer with a proximity type ON/OFF switch with the aid of the assembly instructions and wire it according to the circuit diagram.



Task:

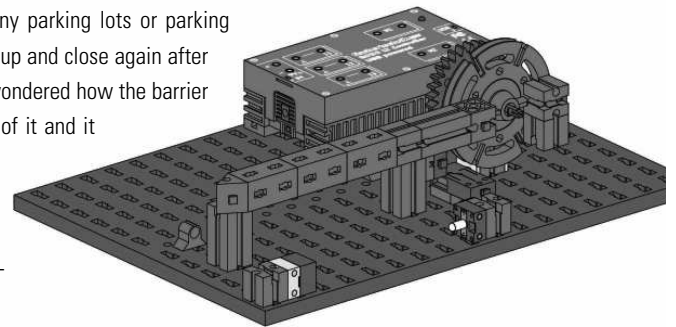
When the light barrier is interrupted, the fan should start running. As soon as the light barrier is no longer interrupted, the fan switches back off.

Finished Program: Hand dryer.rpl

Barrier

- Barriers are located in front of many parking lots or parking buildings. They open when a car drives up and close again after the car drives through. Have you ever wondered how the barrier knows when a car is standing in front of it and it needs to open?

Build the model using the assembly instructions and connect the cables according to the circuit diagram.



Task:

At the beginning the barrier should be closed. Then when the light barrier is interrupted, the arm moves up. After 3 seconds it should close again.



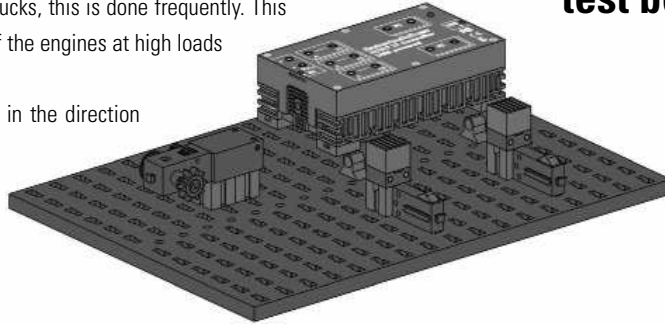
Finished Program: Barrier.rpl



■ An engine test bench allows certain technical data on the engine to be tested or measured. Typical data includes such values as the rpm, temperature or output. In the case of internal combustion engines such as in cars, motorcycles and trucks, this is done frequently. This allows engineers to observe the behavior of the engines at high loads or at extremely low temperatures.

For your motor it is important that it turns in the direction intended by the program.

Build the test bench using the assembly instructions and connect the cables according to the circuit diagram.



Motor test bench

Task:

With your own motor test bench you see whether the connected motor is operating properly. As long as button I1 is pressed, the motor should turn counter-clockwise. As soon as button I2 is pressed it should turn clockwise. The motor should stop when neither of the buttons is pressed. Simultaneously a red light should indicate when the motor is turning counter-clockwise and a green light when it is turning clockwise.



Finished Program: Motor test bench.rpl

■ Machines make our lives easier by accomplishing heavy and dangerous work for us. However machines sometimes pose a danger themselves. Automatic punching machines or presses are typical examples. On the one hand it is necessary to get very close to the machine to insert the parts to be processed. On the other it is imperative to stay away from the machine when it starts operating.

In order to protect machine operators, safety experts have come up with safety devices such as two-hand controls and light barriers.

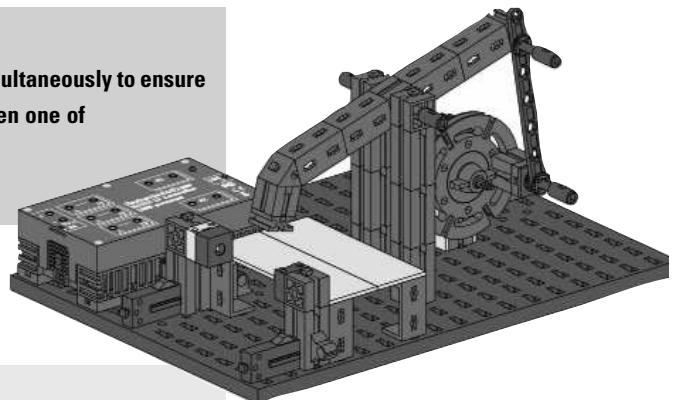
Build the model with the safety devices and connect the cables according to the circuit diagram.

Task:

The machine should start only when both buttons are pressed simultaneously to ensure that neither of the operators hands can be in the machine. When one of the buttons is released or the light barrier interrupted, the machine should stop.



Finished Program: Machine.rpl



Machine

"You can find additional fischertechnik construction sets, with which you can expand your programming knowledge, at www.fischertechnik.de."



If something doesn't work right ...

... you will hopefully find a solution for your problem in this table.

Problem	Possible Cause	Fault Clearance
1. Software ROBO Pro Light can't establish a connection to the ROBO LT Controller.	USB cable is not connected.	Plug in the USB cable.
	USB driver is not installed.	Install the USB driver – see installation instructions, which are contained in the construction set.
2. The push-button switch does not work.	Electrical plugs plugged into the wrong connections of the push-button switch or the ROBO LT Controller.	Use the connections 1 and 3 on the push-button switch. Plug in the plugs into the two jack sockets for I1, I2 or I3 on the Controller.
	For control programs, which you create yourself: In the dialogue box of the program element, incorrect output M1 or M2 selected.	Check control program and set correct input.
3. Phototransistor doesn't work.	Electrical plugs plugged in incorrectly.	On the phototransistor: Plug in the red plug to the side with the red dot, green plug to the side without marking. On the Controller: Plug in red plug to I1, I2 or I3, green plug to the associated connection.
	Lens tip lamp for the light barrier does not illuminate.	Connect the lens tip lamp to M1 or M2 and turn it on.
	The light from the lens tip lamp passes diagonally past the phototransistor.	Move the lens tip lamp so that the light hits the phototransistor.
	For control programs, which you create yourself: In the dialogue box of the program element, incorrect output M1 or M2 selected.	Check control program and set correct input.
4. The motor does not rotate or the light does not illuminate.	Motor or light not connected to the LT Controller.	Connect motor or light to the LT Controller according to the circuit diagram for the particular model.
	Motor or light connected to the incorrect motor output of the LT Controller.	Use the circuit diagram to check which output M1 or M2 is to be used for the motor or the light and connect with this output.
	For control programs, which you create yourself: In the dialogue box of the program element, incorrect output M1 or M2 selected.	Check control program and set correct output.
	The green LED on the ROBO LT Controller blinks very fast (about four times per second): Short circuit on one of the outputs. Plus and minus poles are directly connected.	Check cables and eliminate the short circuit.
5. Motor rotates in the wrong direction.	For electrical plugs, red and green were mixed up.	Reverse red and green plugs on the motor.
		Change the direction of rotation of the motor in the control program.
6. Sample programs for the models cannot be found.	You don't know, in which folder the programs are located.	Sample programs for all models of the construction set are found at: <i>C:\Programs\ROBOPro-Light\Sample Programs\LT-Beginner-Set</i>
7. Problem not described here.	Not found.	Contact fischertechnik directly, for example, at: www.fischertechnik.de