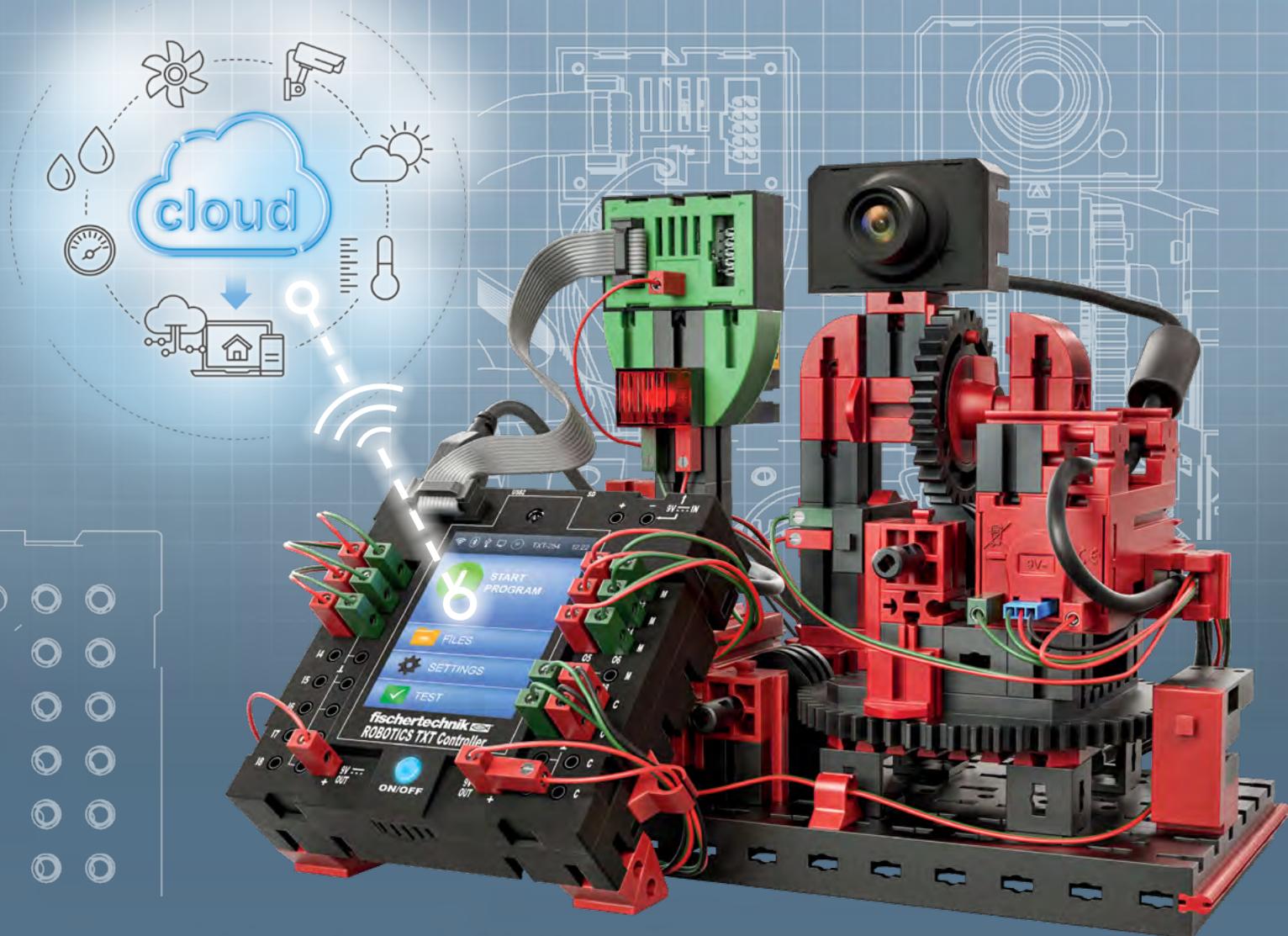




fischertechnik 

ROBOTICS

Begleitheft
Activity booklet
Manual d'accompagnement
Begeleidend boekje
Cuaderno adjunto
Folheto
Libretto di istruzioni
Сопроводительная инструкция
附带说明书



TXT Smart Home
Robotics Sensor Station IoT

6 MODELS

Welcome to the fischertechnik World of ROBOTICS	4
About this Activity Booklet	5
Introduction	6
Digitization	6
Internet of Things	7
Smart Home	7
Industry 4.0	8
Some General Information	10
Electricity	10
Component Explanations	10
Software ROBO Pro 4.4.x	11
ROBOTICS TXT Controller	11
Power Supply	12
First Steps	13
Starting ROBO Pro	14
Light with brightness sensor	16
Photo Resistor	16
LED	16

Comfort Sensor	23
Frost Monitor	24
Mold Monitor	25
Air Quality Sensor	25
Stationary Camera with Environmental Sensor – Data Logging	27
Barometer	35
Sensor Station with Movable Camera	37
Surveillance Camera	37
Cloud Computing	43
Trouble Shooting	49
Interface Test	49
Cables and Wiring	49
Short Circuits	50
Errors in the Program	50
Last Sources for Help	50

Welcome to the fischertechnik World of ROBOTICS

Hello!

With this product “ROBOTICS TXT Smart Home” or “Robotics Sensor Station IoT” you can solve fascinating problems under the motto Smart Home, Digitization and Internet of Things.

Read this digital booklet and complete the individual projects, to learn step-by-step how you can program simple as well as complicated controls for Smart Home projects using the ROBOTICS TXT Controller from fischertechnik.

Learning is a process of building things up from a foundation and it is not possible to start with the most difficult things right away, even though they may be a little bit more interesting than the more simple tasks. This is why we have structured the projects in this booklet so that you learn something different with every new task and can then use this as the basis for the next task.

Your team from

fischertechnik 



About this Activity Booklet

This PDF activity booklet has a number of functions, which are not present in the printed booklet and which may already be familiar to you from the Internet.

Links within the Booklet

- **Links within the Booklet**

When something is mentioned somewhere in the text, which is explained in more detail at another point in this booklet (for example, components), the text appears in dark blue and underlined. You can click on the text to go automatically to the page containing the explanation. This is a so-called "cross reference".

- **Background Infos**

In some cases in this booklet, there are terms or foreign words, which may require explanation. These terms are displayed in green and underlined. If you touch the text with the mouse pointer, a window appears with an explanation.

Introduction

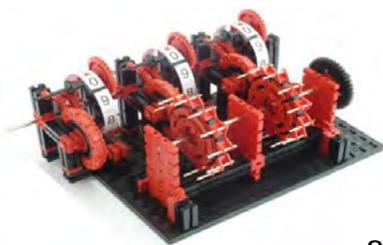
Internet of Things
Smart Home
Industry 4.0
Digitization

Before starting with the models and your programming, I would like to entice you with a small preview of these interesting projects. This also includes a glimpse into the new control-related features with their specific technical names, to be precise, what do we mean by “Smart Home”, “Internet of Things”, “Digitization” and “Industry 4.0”



Digitization

If you check the Internet for the term “Digitization” you will find that the term and the associated technology are actually not all that new. For example, Gottfried Wilhelm Leibnitz invented the first “computing machine” in the 17th century. Another milestone in digitization was Joseph-Marie Jacquard’s loom controlled by punched cards (1805).



Reconstruction of a computing machine designed by Leibnitz in the 17th century

What does digitization mean? Technically speaking it is the conversion of analog data, such as 100 m, 30 seconds, into digital data, consisting exclusively of a series of zeros and ones. The analog number 10, is represented in digital form by the character string 1010. However, digitization is also a means of storing and processing data without using paper. This was initially accomplished by using electron tubes, later transistors and, since approx. 1970 using microprocessors, which are still in use today; naturally in an advanced form.

Can you find any examples of digitization in your surroundings? Do you still read printed books, or do you already have a digital “E-book”? Do you listen to digital radio programs? What about your music CDs, or your computer games? You receive voice messages on your smart phone. A (digital) computer voice reads you the text. The navigation system (GPS) in the car also determines the position of the vehicle with the aid of

digital radio signals, which are then used by a computer to calculate the route to your destination and display it on the GPS screen.

Internet of Things

This is certainly a term you will encounter often in the next few years. But what does the term “Internet of Things” actually mean?

This is primarily about information society technologies, which allow global networking (using computers around the world) of real or virtual objects in such a manner that they can work together and exchange information with one another

You can find examples of this theme on the Internet.

Here, for example, is a description of how a package is identified automatically, allowing the receiver to follow the current status of the shipment.

An intelligent refrigerator is also an interesting example. The user simply enters or reads in a bar code or QR code specifying the content as well as the quantity of a product. When the quantity drops below the specified value, the refrigerator software reminds the user to replenish the product or does it itself automatically.

Smart Home

You may ask why we have named the new ROBOTICS TXT construction set Smart Home? If you look at the models, you can see the reason. All of them deal with the subject of “Efficient Living”. Today it is possible to network all of your household equipment (e.g.



lamps, heating system) or household appliances such as the range or washing machine so that they can be monitored and controlled from one central point. An intelligent heating system can, for example, determine the temperatures in various rooms in your house in relation to the outdoor temperature and use this information to control the heating system.

Industry 4.0

We frequently call this the fourth industrial revolution. Although many people have probably never heard of this term, we have been living with it for some time now.

- It all began at the end of the 18th century with the steam engine. This invention laid the cornerstone for the first industrial revolution.
- At the beginning of the 19th century, electricity was instrumental in introducing industrial mass production. This was the beginning of Industry 2.0.
- Industry 2.0 was superseded by Conrad Zuse's invention – the first computer, called Z1. Introduction of "electronic data processing" heralded in the era of the 3rd industrial revolution.
- Industry 4.0 was first presented as a concept at the Trade Exhibition in 2011 in Hannover, Germany.

Industry 4.0 will change particularly job design in the areas of organization of work, mechanization of work and personalization of work.

Today these 3 factors have already been implemented in the digital factory.

Here is a minor example from the printing sector. A customer wants to make a photo album of her last vacation. For this purpose she first edits the (digital) photographic data on her PC. Then she opens an Internet site for an online print shop, where she selects the proper link, such as Photo Album. Then a program opens allowing the customer to select all the desired design features. For example, the paper, the background color and much more. This program is provided for the user in a cloud.

After entering all the data, a layout is created using the data entered and the price is calculated. When the customer confirms the order, it is processed digitally. This means, the appropriate paper and printing ink are selected and sent to the printing press to complete the order. After printing the pages as well as the back and cover sheet, they are automatically sent on to the binding machine where they are bound. A small robot transfers the finished album to the packaging station. Here the album is automatically packed, marked with the shipping

address and placed in a storage location for shipment. After reaching the shipping department, it is automatically sent to the proper shipping agent.

The first human the album comes into contact with is the driver of the package van as he delivers it to the customer.

But how much longer? Even today we are attempting to use driverless transport systems or drones for logistic support. Will these soon take over delivery too?

Some General Information

Before we really get started with the construction set, you still need to know a few things. Even though the components we will work with are very robust, if you do not handle them properly, they can be damaged under certain circumstances.

After you have unpacked all the building blocks, it is necessary to first assemble a few components such as cables and plugs before you can really get started. Details are given in the assembly instructions under "Assembly Tips." It is best to do this first.

Electricity

As you certainly know, a lot of the components in the ROBOTICS TXT Smart Home Set use electric power. And you know it is necessary to be particularly careful not to make any mistakes when working with electrical components. That is why you should always read the assembly instructions very carefully when wiring the electrical components.

Never connect the positive and negative poles directly to one another to prevent a short-circuit. This can damage the [ROBOTICS TXT Controller](#) or the rechargeable battery.

Component Explanations

The construction set contains all of the following

First, it contains numerous fischertechnik building blocks, as well as motors, indicator lights and sensors and colored assembly instructions for building various models.

Actuators



Actuators are all components, which can perform some type of action. This means that they become "active" in some way when they are connected to electric power. In most cases you can see this directly. A motor runs, an indicator light illuminates and so forth.

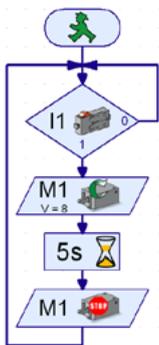
Sensors



Sensors are so to speak the counterpart to the actuators [Actuators](#). This is because they do not perform any actions, but react to certain situations and events. For example, a pushbutton reacts when pressed, allowing an electric current to flow or interrupting its flow. A heat sensor reacts to the temperature in its surroundings.

I will provide you with an explanation of the individual actuators and sensors in the project, when they are used for the first time.

Software ROBO Pro 4.4.x



ROBO Pro is a graphic programming interface for creating programs for the [ROBOTICS TXT Controller](#).

A "graphic programming interface" allows you to compile programs visually with the aid of graphic symbols instead of "writing" them out by hand line for line. An example of such a program is shown at the left.

The procedure for creating such a program is described in detail in the Chapter "First Steps". The [ROBO Pro Help](#) feature also shows how this works in Chapters 3 and 4.

The current software is already installed on your PC.

ROBOTICS TXT Controller



The ROBOTICS TXT Controller is the heart of this ROBOTICS construction set. It controls the [actuators](#), and evaluates the information from the [sensors](#).

For this purpose the ROBOTICS TXT Controller has numerous terminals for connection to the components. The instruction manual for the ROBOTICS TXT Controller describes which components can be connected to which connections and the functions of the connections.

The color touch screen allows convenient operation of your ROBOTICS TXT Controller. The camera contained in the construction set can be connected to the USB host port (USB-1). The integrated Bluetooth and WLAN interface is a particularly interesting special feature. It allows you to complete a wireless link between your PC and the ROBOTICS TXT Controller or between several controllers.

You can define how the controller interacts with the individual components and what they are to do in detail in the program you write with the [ROBO Pro Software](#).

Power Supply



As you know, many of the components in the ROBOTICS TXT SmartHome Set need electricity to operate, so naturally you also need a power supply for the TXT Controller.

The fischertechnik power pack is best suited for this.

It is not included in the construction set.

A Few Tips

Experimenting makes the most fun when the experiments also work. This is why you should follow a few basic rules when building the models.

Work carefully

Take your time and look precisely at the assembly instructions for the model. Looking for an error later will take much longer.

Check the movement of all parts

When putting models together continually check to see if parts, which have to move, move easily.

Using Interface Test

Before starting to write a program for a model, you should test all parts connected to the [ROBOTICS TXT Controller](#), using the interface test feature in ROBO Pro. How this works is described in the [ROBO Pro Help](#) in Chapter 2.4.

First Steps

Now that you have made all of the preparations and read the information, you can finally start working.

This chapter describes how to:

- build the first simple model, a light with brightness sensor, and connect it to the [ROBOTICS TXT Controller](#),
- connect the [ROBOTICS TXT Controller](#) to the power supply and PC,
- create and start your first simple program with ROBO Pro

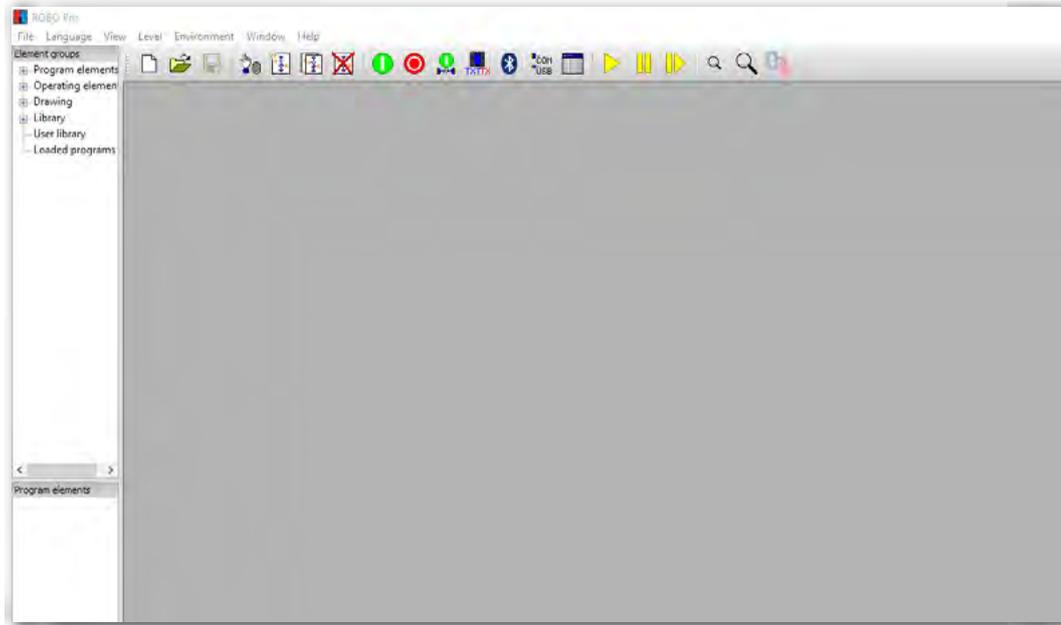


Since you will be working particularly with the [ROBO Pro Software](#) in addition to the fischertechnik components themselves, you should be familiar with the details for writing programs. And because this is explained very clearly in [ROBO Pro Help](#) Chapters 3 and 4, it is best at this point to continue by working through these chapters carefully.

Our tip also applies here: Take your time and concentrate; then you will have much more fun with the models later.

Starting ROBO Pro

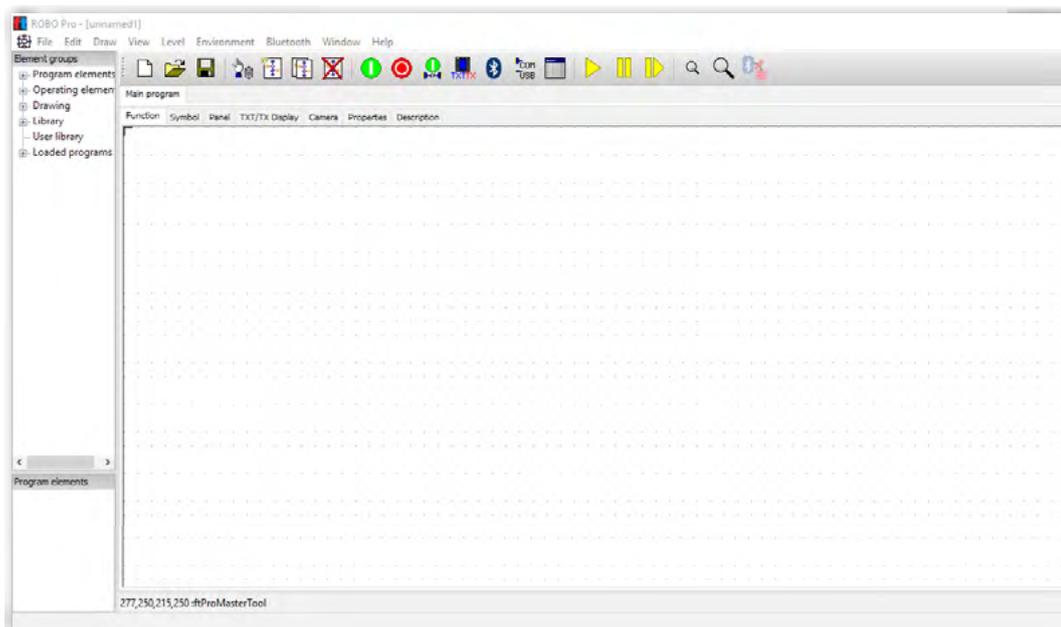
Start the ROBO Pro program. The following working display appears.



New

To create a new program, click on the "New" button with the mouse.

ROBO Pro then switches over to the programming mode. You can tell this, because the program monitor displays a dot matrix.





Save

To save a program click on the “Save” button with the mouse. ROBO Pro automatically switches to Explorer and displays your computer’s table of contents. Here you can select a folder, such as “ft programs” for example. Enter a “file name” to remind yourself what it is, for example “brightness sensor”.



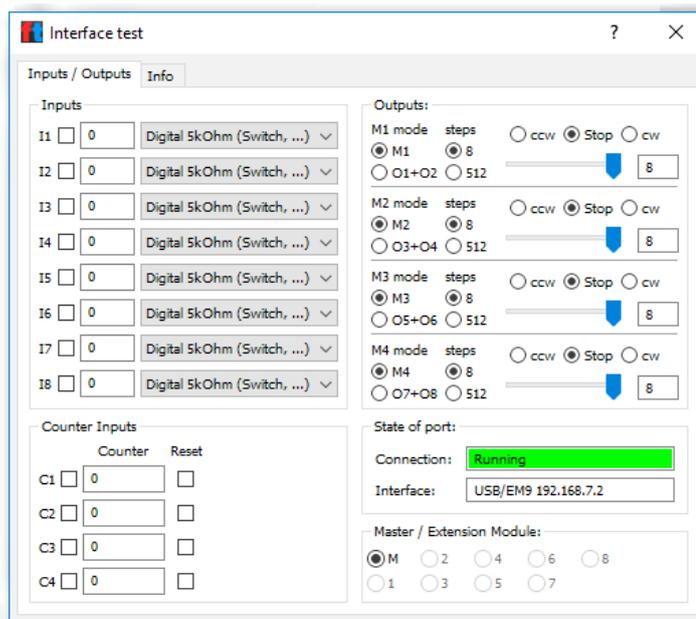
Open

Use the “Open” button to open an existing file. When you click on the button, the program goes to the Explorer on your computer. Here you can select the file you want to load.



Interface Test

It is most practical to check whether the TXT Controller is properly connected to your computer, each time, before starting to work. To do this perform an “Interface Test”. To do this, click on the "Interface Test" button. The following info window appears:



Here all important interface data is displayed virtually. If, for example, you have connected a pushbutton (input) which you then push, the switching status is displayed with a check mark. The “Output” window indicates if you have switched on a motor or light. More information is available at [ROBO Pro Help](#).

Example programs: Complete example programs are available in ROBO Pro for every model in the TXT SmartHome construction set. These are saved under:

C:\Program files(x86)\ROBOPro\Sample Programs\ TXT Smart Home

Light with brightness sensor

The first model you built according the assembly instructions, serves to control an outdoor light depending on the ambient brightness.

To program this, use programming level 1 for your first program.



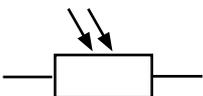
Photo Resistor



ft building block
Photo resistor

Use a photo resistor for the sensor.

A photo resistor is an electronic component, whose electrical resistance changes when it is exposed to light. In many descriptions you will also find the designation LDR. This is an abbreviation for "Light Dependent Resistor".



Circuit symbol

LED

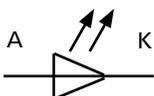


ft building block
LED

Use an LED light as an actuator.

An LED is an electronic component, which converts electrical energy to light. The abbreviation LED stands for "Light Emitting Diode".

Important: The photo resistor has a resistance value of 0 – 5000 Ohms. The interface display indicates a value of **0** at full brightness and **15 000** in complete darkness. For your project it will be necessary for you to experiment a little bit with the figures and intermediate values.



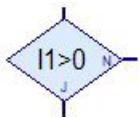
Circuit symbol

Project 1

As soon as it becomes dark, the lights should turn on. It will be necessary for you to find the [threshold value](#) experimentally.



Start the program with the "Green walk light". First you need a program command to "Branch analog" [ROBO Pro Help 8.1.4](#) which checks the resistance of the photo resistor. Drag this to the work screen and drop it below the "Green walk light".

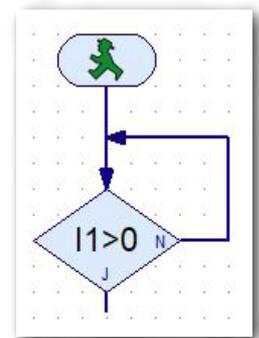


Analog Branch

Information on "drag and drop" and "docking" is given in [ROBO Pro Help 3.3 and 3.4](#).

You will find that this command has 2 outputs. Y for Yes and N for No. You can ask a simple question here.

If the value at input I1 is "> greater than 0" then branch to Y, otherwise to N. Since your program continually checks the [resistance value](#) of the photo resistor, the N output is connected to the input.



First it will be necessary for you to set the type of sensor connected for the branching operation. For this purpose click on the command with the "right mouse button". The context menu for the branching operation appears. Use the mouse to click on the "Down Arrow" button for "NTC resistor". Another context menu appears, in which you can select "Photo resistor".



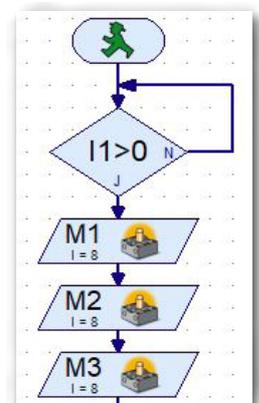
Motor Output

Then insert the program command "Motor output" [ROBO Pro Help 8.1.6](#) in to your program. Since you want all of the lamps to come on, it will be necessary for you to insert this command 3 times.

Here it is also necessary for you to make a few changes. Click on the command with the right mouse button. A context menu opens for setting various parameters.

It is necessary to switch over from "Motor" to "Lamp". In the action field click on the circle in front of "On".

This causes the light to switch on when the previous condition is met (Y). It is also necessary for you to specify the lamp connections, M1 - M3.



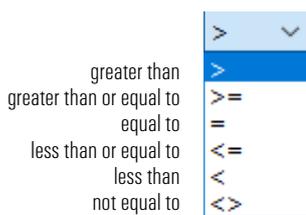
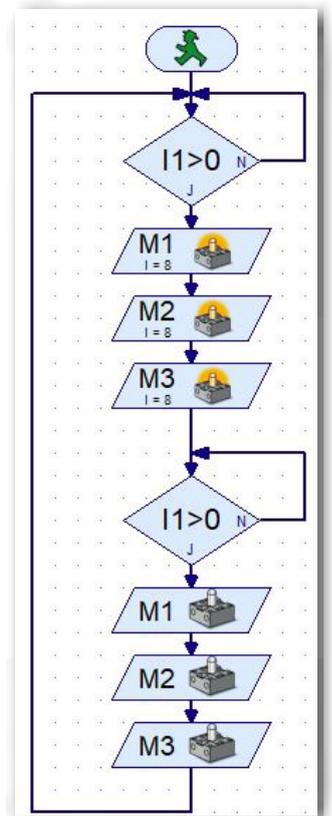
The changes are shown in your program.

Expand your program again with "Branch analog" and 3 motor outputs.

In the context menu for the motor output, change the parameters to "Lamp" and "Off". Change the motor connections to M1 – M3.

Connect the output from M3 to Input I1 (loop)

Before testing the program for the first time, it is necessary to change a parameter for the two branches. Click on the button with the right mouse button. The "Branch" context menu appears.



Make the entry in the "Conditions" field. Two windows are available here. Help is available at [ROBO Pro Help under 8.1.4.](#)

After opening the context menu, you can select the appropriate conditions.

Select ">4000" as the "threshold value" for the first check (LED on) and <3000 as the threshold value for the second check (LED off).

Then you can test the program.



Start program in online mode

To do this click on the "Start" button. [ROBO Pro Help 3.7.](#)



Since the program is in an endless loop, it is necessary to stop it with the "Stop all programs in progress" button.



Save

Does the program work? If so, save the program on your computer. Use a name you can easily remember.

Finished Program: Lighting_1.rpp

Little additional assignment:

Experiment a little bit with the parameters for the threshold values. When does the LED switch on, when does it switch off.

So, we got that finished. Are you ready for your second lighting project now? Again, I will be right here to help you.

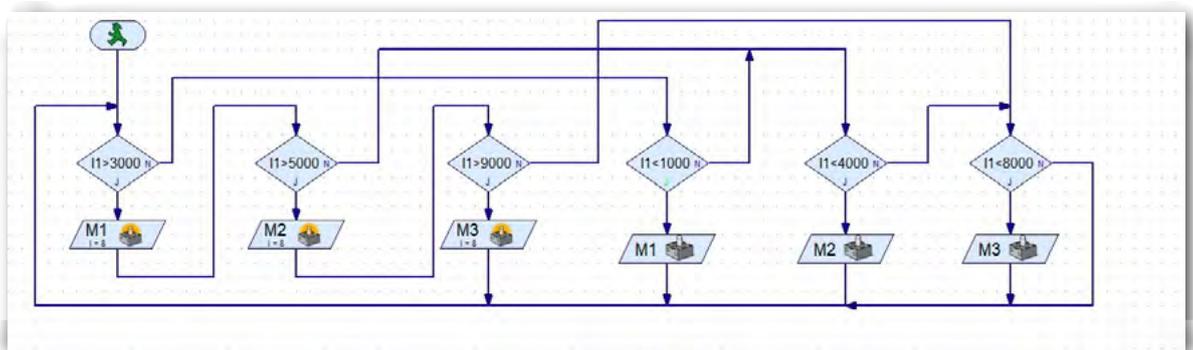
Project 2

The outdoor lighting is changed so that the lights switch on one after another as it becomes increasingly darker until all three lights are illuminated when it is completely dark.



We will complete this project together.

In this project you will have to poll the sensor at I1 six times, to get the three lamps to switch on and off at the values supplied by the photo resistor.



What happens in this program sequence? First the sensor value is checked for lamp M1. If the value is "Yes", the program goes to lamp 1 and switches it on. The program then continues by checking the value for the second light via the output for M1. If this value also corresponds to the check, the second light is also switched on. This also applies for the third light.

Let's look at the three N outputs for the first three checks. If three values are negative, i.e. N, the program branches to the three checks for switching off the lights. If the values here also correspond to the values for switching off the lights, the program switches off the light in

question. At output “N” the program returns to the starting point and restarts the switching operations.

Does the program work? OK, save it on your computer.

Finished Program: Lighting_2.rpp

Noise sensor

You can build the second model as described in the assembly instructions and wire it accordingly. Use Level 3 in ROBO Pro for programming. This provides you with significantly more commands for programming.



Actually in this project you only need to replace the photo resistor with the camera, because the required sensor is already in the camera housing.

The microphone, installed on a circuit board in the housing, serves as the sensor.

Microphone

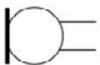


Microphone

The microphone used in your camera is an acoustic (sound) converter for converting acoustic vibrations in the air into electrical voltage. The variations in this voltage can be measured and processed. The unit of measure for sound is “Decibels”.



You can find further information on the corresponding page in Wikipedia. Here you can find out about the various designs and functions.



Circuit symbol

Project 1

Create a program that continuously measures the noise level. At a certain volume of 55 decibels, the green light illuminates. At a volume of 65 decibels, the yellow light illuminates. And at a volume of 75 decibels, the red light illuminates.



Generate new subroutine

You will use a "subroutine" for the first time in this program. You can create the subroutine by clicking on the command "Create new subroutine".



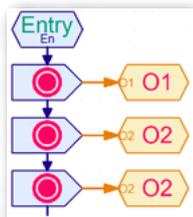
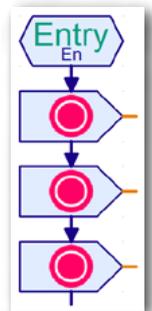
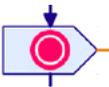
In the same manner that every main program starts with a green walk signal, a subroutine also starts with a corresponding command. [ROBO Pro Help 8.3.1 and 8.3.2](#)



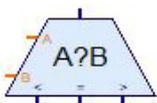
The "subroutine" block comprises two additional commands. These serve to read information in to and out of the subroutine [ROBO Pro Help 8.3.3 and 8.3.4](#).



First insert the subroutine input into the function window. Then insert a new command "Off" [ROBO Pro Help 8.5.8](#) from the "Commands" block. You will need it 3 times (3 lamps).

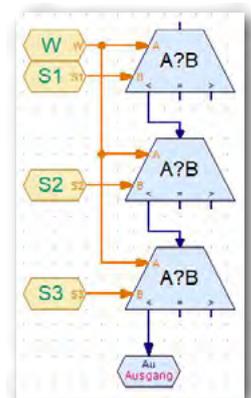


Dock one output command on each of the orange colored outputs. Right-click on the context menu to change the names to "01 - 03".



To evaluate 3 acoustic values, you need the next command, "Compare" from the "Branch, Wait ..." block [ROBO Pro Help 8.3](#).

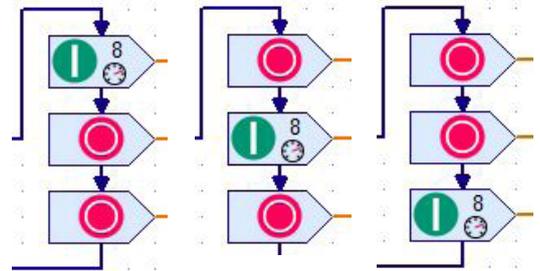
You will need to dock a input command from the "subroutine" block on each of the two inputs. Change the inputs for the first comparison command to "W" and "S1" using the context menu. Connect each of the other two comparison commands to input "A". Change the two "B inputs" to "S2 and S3". Then connect the "<" output



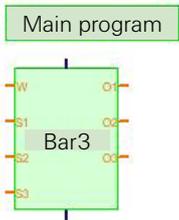
to the "Compare" input. Do the same thing with the second "Compare" command. Connect the last output for the third comparison to the output command.

Then insert the assignments (On/Off) into the program.

Connect outputs "=" and ">" to the lamp assignment inputs. Connect the end of the lamp assignment to the input for the next "Compare" command.



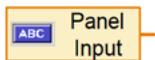
Connect the orange outputs accordingly to outputs "01 - 03".



Now insert the subroutine into your main program. To do this open "Loaded programs" and there, for example, "Noise sensor" Two green blocks appear in the program element field. You need the second block.



Dock the command "Lamp output" from the "Inputs, Outputs" block on each of the outputs [ROBO Pro Help 8.7.5](#). Rename them "01 - 03".



Then insert the command for value input. For this purpose select the command "Control Field Input" from the "Inputs, Outputs" block [ROBO Pro Help 8.7.5](#). Switch over to "Mike" and "Logarithmic:" in the context menu



Insert the command "Constants" from the "Variable" block at connections S1 - S3 [ROBO Pro Help 8.4.3](#).. Change the input values in the applicable context menu to S1 = 55, S2 = 65, S3 = 75.



Wait

Insert the command "Wait Time" from the "Basic Elements" block at the subroutine output. [ROBO Pro Help 8.1.5](#). Change the value to 0.01 seconds. Connect the command output to the program input.

The program is now finished and ready for testing. If everything is OK you can save the program on your computer.

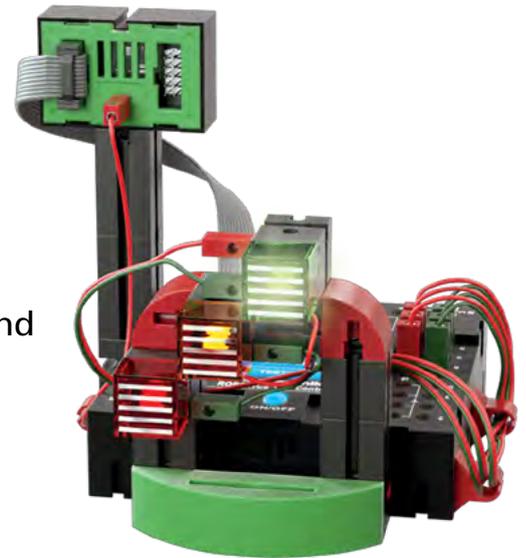
Finished Program: Noise_sensor.rpp

Comfort Sensor

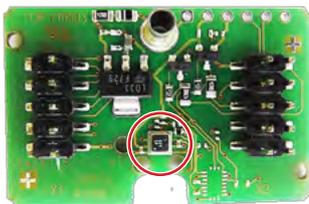
Build the third model as described in the assembly instructions and wire it accordingly.

Here it is only necessary to replace the camera with the environmental sensor and raise it with two 30 building blocks.

In the individual projects you will use it as frost monitor, a mold monitor and air monitor.

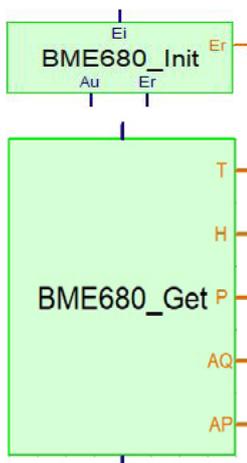


Environmental Sensor



The sensor, located on a circuit board in the housing, serves to measure gas, atmospheric pressure, humidity and air temperature. The module can be connected to the [TXT-Controller](#) using a ribbon cable.

We have saved a finished block for evaluating the sensor data under: “



Element groups – Library – I2C – Environmental-Sensor-BME680

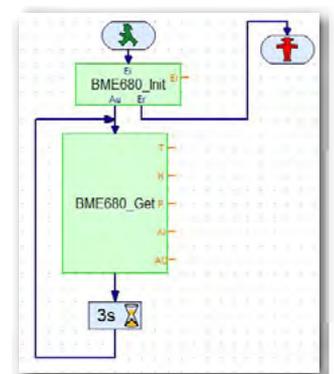
The block consists of 2 blocks: the Init Block and the Get Block.

The Init Block starts all processes for evaluating the polling operations. The Get Block requests data from the sensors for processing in another program.

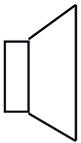
The output designations stand for

- T = Temperature (°C)
- H = Humidity (relative humidity in %)
- P = Atmospheric pressure (hPa for Hecto-Pascal)
- AQ = Air quality (readings from 0 – 500)
- AQP = Reliability of AQ values for readings over a certain period of time (reading 0 – 3).

First compile the program shown at the right and then add the other commands.



Frost Monitor



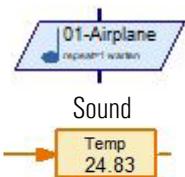
Circuit symbol

For acoustic signals use the loudspeaker integrated into the TXT-Controller.

Loudspeakers convert an electrical input signal into mechanical oscillation of the loudspeaker diaphragm. These oscillations are perceived as sound. A loudspeaker consists of a housing, a permanent magnet and a moving diaphragm.

Project 1

Create a program that continuously measures the ambient temperature. The red lamp should come on, when the value drops to below 3 °C. The alarm should sound once.

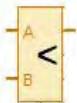


You can use the command “Sound” to generate an alarm tone. You can select various tones or songs in a context menu. [ROBO Pro Help 8.1.12](#).

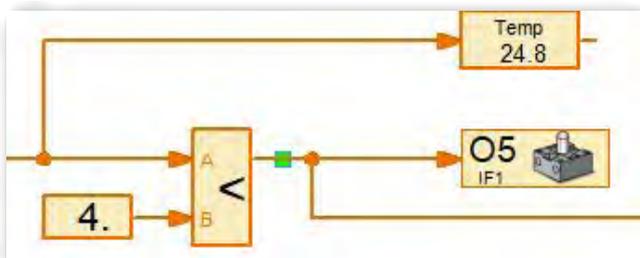
To evaluate the temperature use BME680_Get output “T”. Connect this to a variable command for testing. When the program is started, the current temperature is displayed.



To switch on the sound you need a green walk signal followed by a command “Wait For ...” from the “Branch, Wait ...” program element block. [ROBO Pro Help 8.6.5](#). Then dock the sound command and insert a loop at the beginning of the program.



Insert the element “Constant” at “Input B” Set the constant value to “4” in the context menu, to give an alarm when the temperature drops to below 3 °C.



After completing the program, you can test it and save it on your computer.

Finished Program: Wellness_sensor_1.rpp

Mold Monitor



Mold monitors as well as humidity monitors are used to measure the moisture content in the air, for example, in the basement. They are also frequently placed next to the washing machine or dishwasher to detect water leaks.

Project 2

Create a program that continuously measures the humidity. The red light should illuminate as soon as the humidity has increased to above 80 % (danger of mold forming).



This project only requires you to rewrite the previous program. The check is connected to BME680_Get output "H". Label the variable for displaying the humidity "Humidity" Assign the constant a value of "80" for the humidity threshold.

After completing the program, you can test it and save it on your computer.

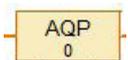
Finished Program: Wellness_sensor_2.rpp

Air Quality Sensor

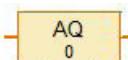


Today this theme is more topical than ever everywhere in the media. Cars, factories, large animal husbandry enterprises, etc. are responsible for the air quality not being as good as it should be.

Many cities have measuring stations to continuously record the air quality. Have you ever seen such a measuring station where you live? With our sensor you can simulate such a station and measure real-time values for the air in your environment.



Reading 0 - 3
0 unusable
3 very reliable



Reading 0 - 500
0 very good
500 bad

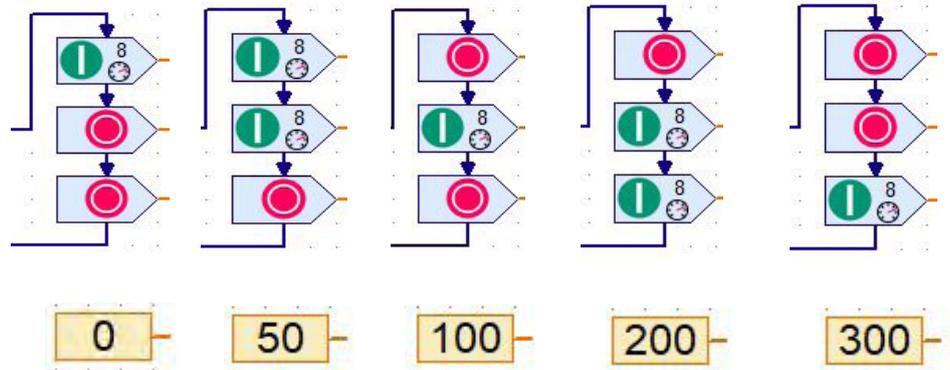
Project 3

Create a program that continuously measures the air quality. You want to indicate the air quality using lights that illuminate differently. The light at output O3 should flash when the value at BME680_Get output AQP drops to < 1. The values should also be indicated on the controller display in the download mode.



The LED´s display the following increments:

Green	very good
green/yellow	good
yellow	satisfactory
yellow/red	sufficient
red	bad



LED bar should illuminate at following readings.



To display the values on the TXT display you will have to add two control field outputs to your program. [ROBO Pro Help 8.7.7](#).

Next switch on the "TXT/TX Display". Add two displays from the display folder. Open the context menu (right mouse button) and change the displays from "AQ = 0 and AQP = 0".

Then link the two control field outputs to the two displays. Switch over to the function window. Right-click to open the context menu. Select "AQ" on the display. The selection is depicted in the command. Continue in the same manner for the second display.

After completing the program, you can test it and save it on your computer.

Finished Program: Wellness_Sensor_3.rpp

Stationary Camera with Environmental Sensor – Data Logging

Stationary cameras are located in train stations, airports, etc. where they serve for surveillance. They are intended to deter crime as well as help investigate crimes already perpetrated.

Build the fourth model as described in the assembly instructions and wire it accordingly.



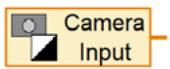
Project 1

Camera as Motion Sensor. As soon as the camera registers motion, an acoustic and visual alarm is output.



In this project all you need to do is check the camera, to see if anything moves in its recording window.

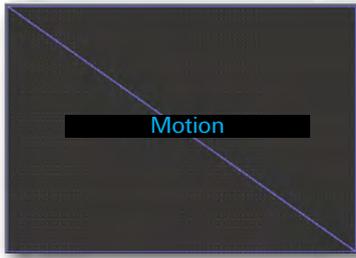
To do this you need the command “Camera Input” from the “Inputs, Outputs” block [ROBO Pro Help 8.7.8](#). Dock this command on the input to the “Branch” command. Connect the lamp command “Off” to the “N output”. and the lamp command “On” to the “J output”. Follow this with the



sound command from the “Basic Elements” block [ROBO Pro Help 8.12](#). Naturally the program runs in an endless loop.



After completing the program, you will have to decide what you want the camera to react to. For this purpose, switch on the menu point “Camera”. An entry window appears for setting various parameters. Important here is “Sensor Fields” from the element groups.



In the program elements appearing, select “Motion”. Click on this command. When you click with the mouse in the dotted window, the cursor turns into a crayon. Use it to draw a frame enclosing the entire field.

Finally it is necessary for you to assign the camera input to the motion monitor. To do this change back to the function window. Click on the input command with the right mouse button. Select the sensor field “Motion A” in the context menu which appears.

After completing the program, you can test it and save it on your computer.

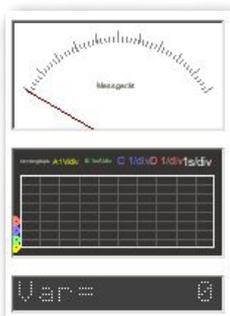
Finished Program: Stationary_camera_1.rpp

Simple Data Logging

The next project will have you log various data. A simple exercise will introduce you to this subject.

Project 2

This project registers the values from the brightness sensor, built into the model, and displays them graphically on an oscilloscope monitor.



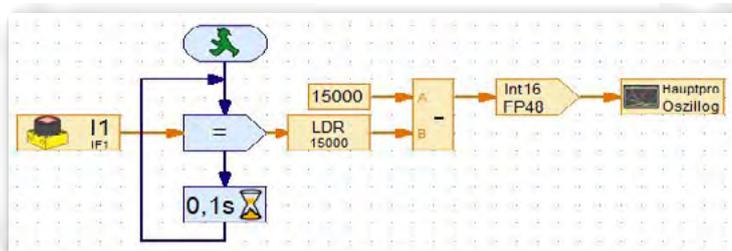
Switch over to “Control Field” to include the oscilloscope monitor in the project [ROBO Pro Help 9.1.4](#). The command field window appears: Click on “Displays” in the “Controls” block

Add the oscilloscope monitor to the window by clicking on the command. The cursor turns into a crayon. Use it to extend the oscilloscope monitor to form a large rectangle. Open the context menu for the oscilloscope and initially assign the name “Oscilloscope”. Define the settings for the readings as follows:



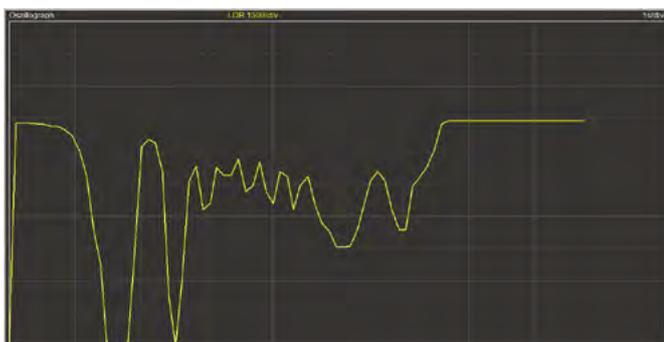
In the "Time and horizontal control" field, change the values for "Y increments" to 10 and "H and V pixels/increments" to 100 um. Confirm these entries with OK. Next write the measuring program.

The photo resistor values are logged using the "Universal Input" command from the "Inputs, Outputs" block. In the context menu change the display to photo resistor. The value is applied to the command



block "Assign" from the "Commands" block [ROBO Pro Help 8.5.1](#). In the context menu change the display to "=". The reading is then assigned to a variable. In the context menu

change the display to "LED". This command follows the command "Arithmetic Operations" from the "Operators" block [ROBO Pro Help 8.8.1](#). Dock the variable output on input "B". Dock the command "Constant" on input "A". Assign it a value of "15,000" (max. photo resistor input value). The output value is the difference of "A - B". This value is forwarded to the command "Calculate with decimal numbers" from the "Operators" block [ROBO Pro Help 13](#). Finally dock the command "Control Field Output" from the "Inputs, Outputs" block [ROBO Pro Help 8.7.7](#). In the context menu, change the display to "Oscilloscope".



Then you can test the program. Switch over to the "Control Field" and start the program. The readings are now displayed on the oscilloscope. Change the brightness by covering the sensor with your hand. The data changes and the oscilloscope indicates this graphically.

Now that we have that finished, you can start working on the next project.

Finished Program: Stationary_camera_2.rpp

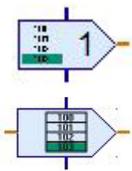
Project 3

This project makes the logged values available in a spreadsheet program, where they are displayed using a beam diagram, for example.

For this purpose it is necessary to save the data in a *.csv file.



This requires expanding the program with two new commands.



Following the “=” command, insert the command “Attach Value” from the “Commands” block [ROBO Pro Help 5.10](#). In the context menu, switch on “Data input for command value” and “48 bit floating decimal”. Connect the data input to the input to the oscilloscope.



Dock the command “List” from the “Variable, Timer, ...” block at the S-input to the orange output [ROBO Pro Help 8.4.5](#).

It will also be necessary for you to make various changes to the list command. Activate the context menu.

- Activate “Write to CSV memory” (check mark).
- Switch over to “48 bit floating decimal”.
- Switch over to “Global”
- Switch over to “How list data”

Define a maximum size of 32767 for displaying the data.

8711
8838
9040
9070
9197
9064
9017
8301
6847
7652
9070
9250
9247
6623
5997
9076
8275
6891
8890
7111

Now start your program. If everything functions properly, the graphic readings appear on the oscilloscope. Quit the data logging. End of program.

Click on File and there on "Save *.csv memory for lists".

Save the data in the window "Data set list output". If a file is not already present, create one in the field "File name".

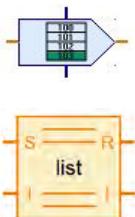


Then open the spreadsheet program. There open the file, for example Data.csv". The readings are listed in column A. In the diagram mode you can display the data graphically.

Finished Program: Stationary_camera_3.rpp

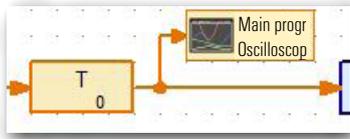
Project 4

Create a program to log the values from the environmental sensor and display them on the oscilloscope in the ROBOPro control field. Simultaneously the values are saved in a *.csv file for graphic display in a spreadsheet.



Use the subroutines BME680_Init and BME680_Get again for this program. Dock one variable command on each of the outputs. These are renamed using the initial designations in the BME680_Get subroutine. Then dock the command "Attach Value" on each of the variable outputs. Do this with each variable. This is followed by the list command, as in the previous program.

You can name the lists to correspond to the previous variables.



Add a control field output to each “Variable - Attach Value” connection. Change the display to “Oscilloscope”. At the end of the program also add a 3 second wait command. Then close the loop with the input to BME680_Get.

Change over to the control field monitor and create your oscilloscope here. It should display 5 values.



Activate the context menu for the oscilloscope. Here you will have to change a few parameters. Take the values from the following table. First assign a name you can remember such as “Oscilloscope”. Switch on the first 5 channels. Change the channels as shown in the table (see Page 23 “Name”).

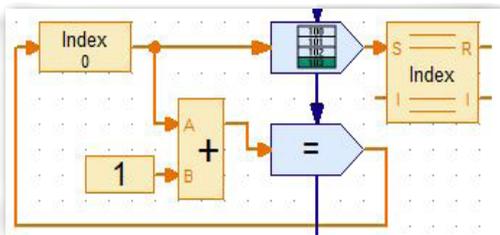
Color	Name	Units	Position	Zero point	Scaling	Auto
red	T		0	20	20	on
blue	H		1	0	41.4	on
green	P	hPa	2	1e+03	1e+03	on
white	AQ		3	0	70	on
magenta	AQP		4	0	1	on

In the function window, activate the context menu for the command “Universal Output” and make the first entry “OscilloscopeT”. Proceed in the same manner with the other outputs (Oscilloscope ...).

Important: Switch over to “48 bit floating decimal” in the “Assign Value” commands. The same applies for the list commands as well. Here, use the changes from Project 3.

Also change the values for “Time and horizontal control”. Again, take the values from the table:

Time/ increment	1s			H=V off
H increments	10	H Pixel/ increment	50	auto on
V increments	8	V Pixel/ increment	50	auto on
Auto-scaling	2.0	Logical increments	0.5	



When you look at the program, you will see, that the BME680_Get output and the input to the first “Attach value” command are still open.

Insert the value assignment and list at the side here. Change the names accordingly to “Index”. This counts how many times the program has run through in “Whole numbers”. Later the readings can be assigned to the index.

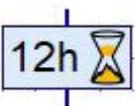
What happens up to this point when the program is started? The readings from the sensor are read into the command “Attach Value” and then saved in the list. Simultaneously the values are displayed on the oscilloscope. The index is incremented by 1 for each run.

Then you can test the program. The readings are displayed on the oscilloscope monitor. Save the data as in Project 3.

Finished Program: Stationary_camera_4.rpp

Project 5

A flower vase with soil and a seed are placed in front of the camera. Create a program to measure the temperature, humidity, atmospheric pressure, air quality and brightness. The values are to be measured every 12 hours and saved in a *.csv file. First take a picture during the day providing information on the growth.



Here you can use the program, "Stationary_camera_4.rpp". It is only necessary to change the wait time in the loop to 12 hours.

You can use the program "Stationary_camera_3.rpp" to read in and display the brightness value.

To take the picture change over to "Camera" in the function list. Here activate the function "Switch on camera". To take a picture, click on "Snapshot". A picture is taken. Simultaneously a context menu window appears for entering the location for saving the image and the file name.

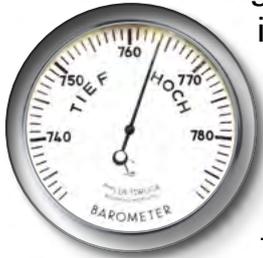
The storage location and file name are used for the other ROBO Pro images. ROBO Pro automatically adds the numerical value.

You can use the readings and images later in your documentation. You can use the individual images, for example, to make a movie, in which the readings can also be integrated.

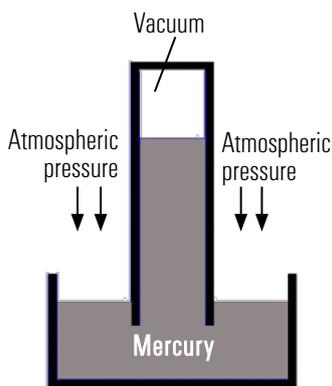
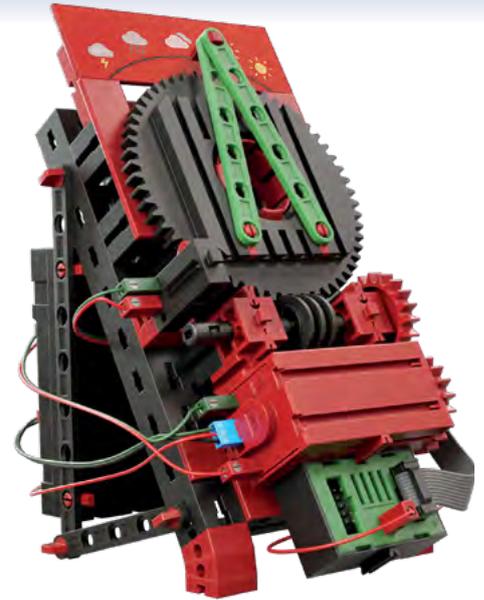
After completing the program, you can test it and save it on your computer.

Finished Program: Stationary_camera_5.rpp

Barometer

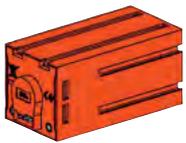


The barometer, an instrument for measuring atmospheric pressure, was developed in its original form around the year 1630. However, it was the middle of the 19th century before glass blowers, opticians and watchmakers succeeded in constructing the first properly functioning instruments. Around 1870, meteorologic designations were added to the scale (Rain, Fair, Change, etc.)



The figure shows a schematic of a liquid barometer. It consists of a tube sealed at the top with its bottom immersed in a supply reservoir. The intrinsic weight of the liquid in the tube causes it to flow downward, out of the tube. This results in a vacuum at the top of the tube. The pressure of the atmosphere counter-acts the downward flow. Thus the column of liquid reaches a state of equilibrium at a certain point.

Construct the model as described in the assembly instructions and wire it accordingly.



Use the encoder motor and worm gear, present in the construction set, to turn the pointer. When the encoder motor turns, it moves the pointer. The motor has a three-pin connector next to the two sockets for the fischertechnik plugs. Counting pulses are output here. The connections C1 and + on the controller are used.

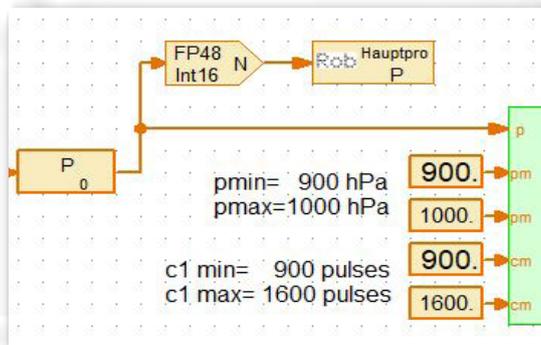
Project 1

When the barometer is put into operation the pointer should first move to the 0 point. The limit switch in the model serves for this purpose. The pointer moves to the weather symbol corresponding to the current atmospheric pressure depending on the atmospheric pressure measured.



For this project it is necessary to convert the reading at BME680_Get output P to counting pulses for the encoder motor.

Dock the following program excerpt on output P.



The values logged in the subroutine "Calculate_Pos" are processed. The pointer position is calculated in the subroutine "PosX".

Before displaying the currently measured atmospheric pressure, the point needs to move to the 0 point (pushbutton). This is accomplished in the subroutine "PosEm". When this is closed mechanically by the pointer, the program moves to the actual measured point and indicates the currently measured atmospheric pressure or applicable weather symbol.

The atmospheric pressure is measured anew every 3 seconds. If the atmospheric pressure changes, the new value is checked by the subroutine PosEMD. If the atmospheric pressure increases or decreases, the program moves the pointer accordingly.

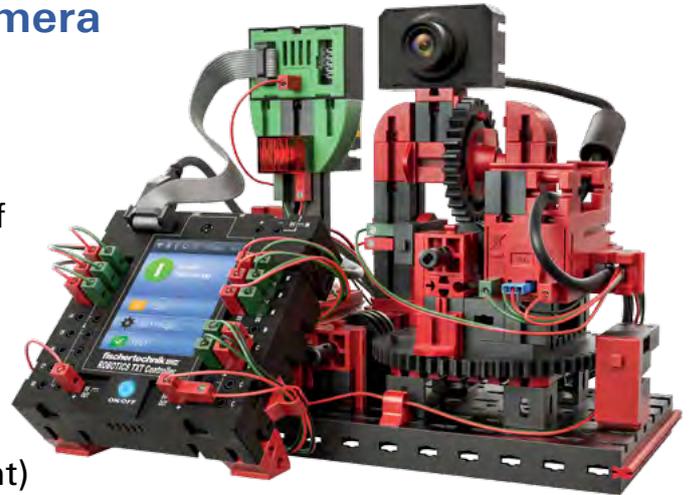
After completing the program, you can test it and save it on your computer.

Finished Program: Barometer_1.rpp

Sensor Station with Movable Camera

This model uses various forms of data transmission.

- Connection via USB cable.
- Wireless via WiFi "Computer – [TXT-Controller](#)" (Access Point)
- Wireless via WiFi "Computer – Router – TXT-Controller" (WiFi Client)



Construct the model as described in the assembly instructions and wire it accordingly.

Important: Pay particular attention to the cable routing, as shown in the assembly instructions This ensures that the cable is not in the way when the camera swivels.

Surveillance Camera

Project 1

This model serves as a surveillance camera. The image from the camera appears in the ROBO Pro control field. The red light comes on. The camera is controlled by the arrow keys in the control field.



First create the control field. To do this switch over to "Control Field" in the program. You will need "Displays, Controls and Camera Display" from the "Controls" menu.



Then drag the camera display into the control field. It is displayed with a certain size. if you want a larger image, click on the display with the mouse. It is then framed in red. Select "Draw" and "Edit". A crayon appears. Click on the display again. It now has blue markings You can increase or decrease the size of the frame by clicking on the bottom right mark and holding down the mouse button.



Next add "Text Display" from the "Displays" block twice [ROBO Pro Help 9.1.2](#). For the first display enter "M1" for ID/Name and "M1=0" for Text in the context menu. In the second display the ID/Name is "M2" and the Text is "M2=0".

Button

Under both displays insert 5 "Buttons" from the "Controls" block [ROBO Pro Help 9.2.1](#).

Arrange the buttons as shown in the figure. Assign each proper designations in the context menu.



When you open the example program, it first runs through the subroutine PTU_Ini. This subroutine moves the camera to a zero point for the x and y axes. Then the programs waits for you to actuate a button in the control field. If you press the L button, for example, the program goes to the subroutine Pan_L. The motor continues to rotate until the L button is pressed.

After completing the program, you can test it and save it on your computer.

Finished Program: Sensor_station_1.rpp

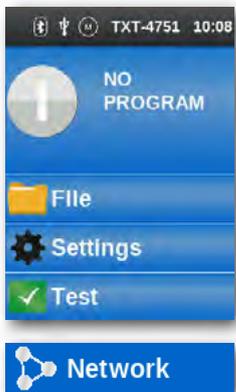


Example of a WiFi router

The next thing you will want to do is control the model via WiFi. The method for creating the connection “Computer – TXT-Controller” as an access point, is described in the instruction manual for the TXT Controller under “Connecting the TXT Controller to the PC via WiFi”. The disadvantage of this connection is the short transmission distance – approx. 10 meters (33 feet).

With the WiFi client the data transmission has a much greater range, depending on the transmission power of the router.

At this point I would like to briefly describe how to create such a connection.



First note the WPA2-PSK address of your router.

Switch on the TXT Controller. Ensure that you do not have a USB connection. Start the ROBO Pro program. Make the next settings initially on the TXT display only.

Switch to “Settings” and select “Network”.



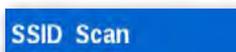
Then select “WiFi Settings” on the monitor.



In the WiFi settings window you need “WiFi Mode”.



In the new window, activate WiFi Client (green dot) and then “WiFi Client Setup”.



Switch on “SSID Scan” Your controller than finds all routers in your surrounding area and displays them in the “Select SSID” window. Click on your router in the list. The window returns to “WiFi Client Setup”.



Your router is displayed under “SSID”.



The next step is to define the “Key” To do this activate “Key”. When the next window appears, activate the entry “WPA2” (green dot) and then “Set Key”



Now enter the “WPA2 address” on the number pad Confirm the entry with the check mark at the top right. The code is displayed with dots in the brackets.



Use the arrow key at the top left to go back until you again reach the “WiFi Mode” display window. Click on the green arrow. The message “Restart WiFi” appears.



The WiFi symbol in the status line and the IP address show you whether the connection was successful.

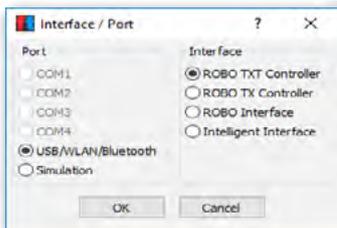
Note the IP address (Client: xxx.xxx.xx.xxx“. You will need this to establish a connection to the TXT Controller.

Now all the settings have been made, and you can return to the display input window.

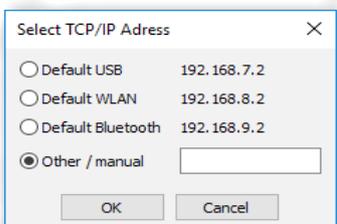
It is necessary to make a final setting in ROBO Pro. You have already started the program.



Activate the button “COM/USB”.

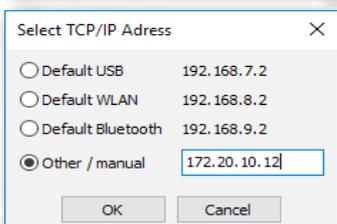


The “Interface” context menu appears. Here “USB/WLAN/Bluetooth” and “ROBOTICS TXT Controller” should be activated. Confirm with OK.



Another context menu appears. Here, switch over from “USB Standard” to “Other/manual”.

Enter the “IP address” for the controller in the rectangular field. Finish the entry with OK.





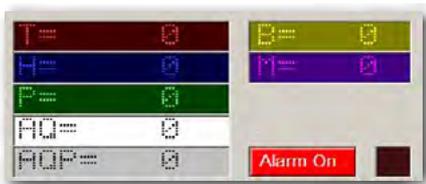
Test: Connect a pushbutton to your computer at I1. Open "Interface Test". Press pushbutton I1. A check mark should appear at I1 on the interface monitor.

Now that you have set up the data link, you can try it out.

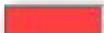
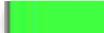
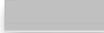
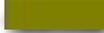
To do this place the model in a different room. Use the arrow buttons in the control field to operate the camera.

Project 2

This model serves as a surveillance station. In addition to controlling the camera, the environmental data is also displayed on the oscilloscope. As soon as the camera registers motion, an alarm is output.



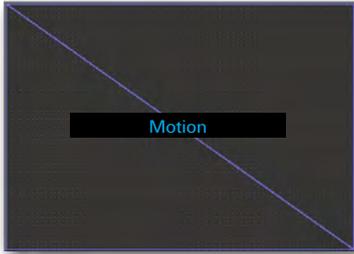
You can use Project 1 and expand it accordingly. On the camera monitor you have 7 more displays and a control. Assign the following entries to the displays in the context menu.

ID / Name	Text	Background color	Text color
T	T=0		
H	H=0		
P	P=0		
AQ	AQ=0		
AQP	AQP=0		
B	B=0		
M	M=0		

Add a control. Insert the labeling text "Alarm On" in the context menu. Set the button color to "Red" and the text color to "White".



Then inset another display next to it. In the context menu enter "Alarm" for the ID/Name, set the color to "Red" and "On at Start" to "Off".



Important: Since the camera is supposed to react to motion, you will have to change the sensor field over to "Motion" as in the project "Stationary Camera with Environmental Sensor – Data Logging".

Add an oscilloscope window next to the Pushbutton and Camera Field. Switch on 7 channels to display the readings. For the first five channels take over the settings from the project "Stationary_Camera_3".

Color	Name	Units	Position	Zero point	Scaling	Auto
yellow	B		5	0	97	one
magenta	M		6	0	82	one

"B" stands for photo resistor, "M" stands for microphone.

You have now completed the entire program, however, for this project you want to display only the environmental data on the oscilloscope.

All other parts of the program will be used in the next project.

After completing the program, you can test it and save it on your computer.

Finished Program: Sensor_station_2.rpp

Cloud Computing

In the next chapter we want to show you how to work with a cloud. The objective is for you to control the “Sensor Station with Movable Camera” model via a cloud connection.

But what does “Cloud” or “Cloud Computing” actually mean?

What does computing have to do with “clouds”? The term means that an IT infrastructure is made available to a user as a service. This can be user software, computing power or memory space. It also means that the IT infrastructure can be installed at the service provider’s, rather than locally on your computer.



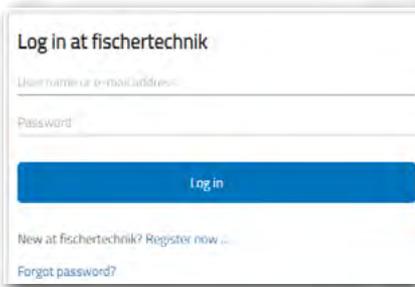
Project 3

The objective is to connect the sensor station to the cloud and display the data there on the dashboard. Any motion detected by the camera triggers an alarm.



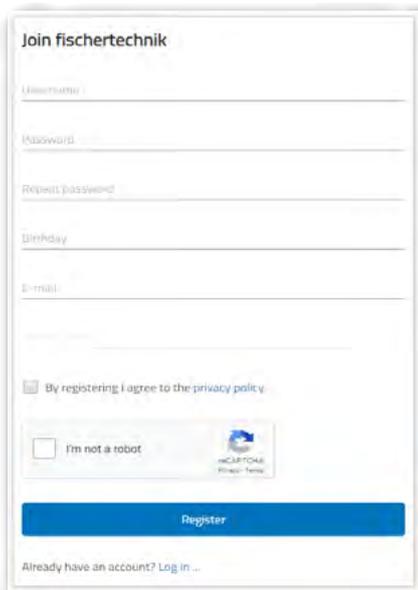
Before you can start the program, you first have to connect to a cloud, similar to connecting to your WiFi router. The next steps show you how to proceed.

Start your Internet browser (preferably Chrome or Firefox) and enter the following address:
www.fischertechnik-cloud.com

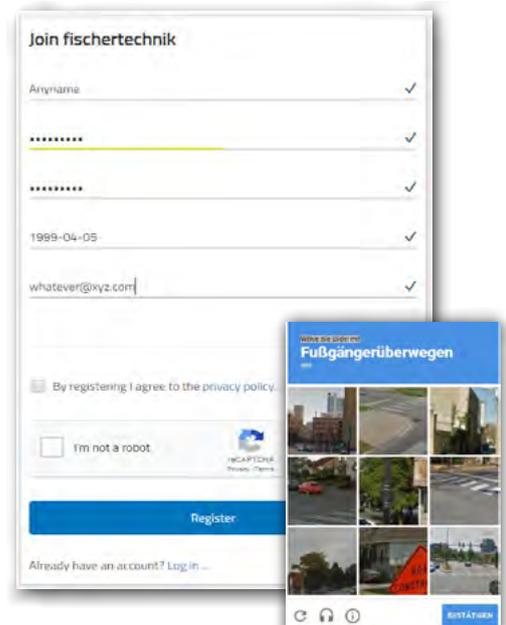


Here you can register your personal access for the fischertechnik cloud. To do this first go to "Register" in the registration window and enter the required data.

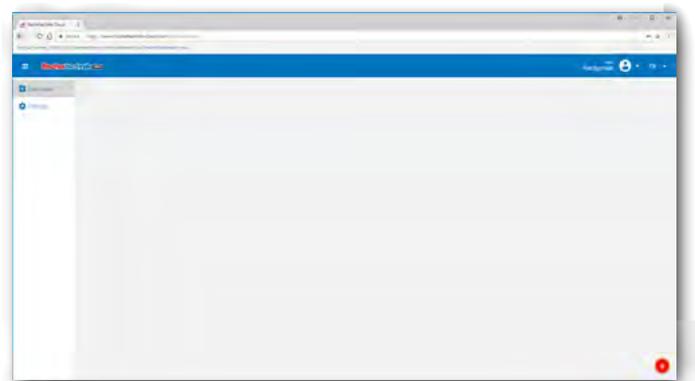
Ensure that you spell everything correctly. It is easiest when you jot down the data.



Important: If you are still a minor, you will need your parent's email address. You need their permission to activate your cloud access. Before clicking on "Register", you need to click on "I am not a robot". A window appears requesting you to click on certain pictures. Complete this selection with "Confirm".



If you want to work with the cloud later, enter the Internet address. The window for registering with fischertechnik appears again. Enter your email address and your password here. Your "Dashboard Monitor" appears.



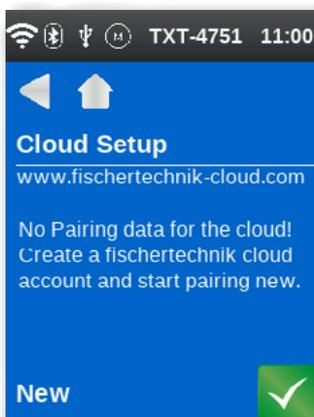
Make the next settings on your TXT Controller.

Important: Check which operating system version is installed on your TXT Controller. The version should be 4.4.1 or higher. This information is available in the menu "TXT / Settings / Info".

Connect your TXT Controller to a WiFi network (see Page 39) as WiFi Client "Settings / Network / WiFi Settings / WiFi Mode".
Recommendation: Set DHCP.



Next activate "Settings" on your TXT Controller and then activate "Cloud Client".

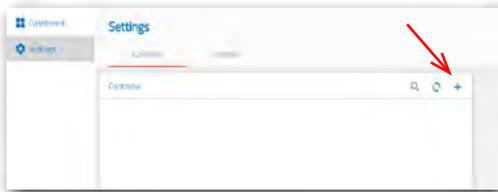


Connect your TXT Controller to the fischertechnik Cloud with "Settings / Network / Cloud Settings / New Pairing".

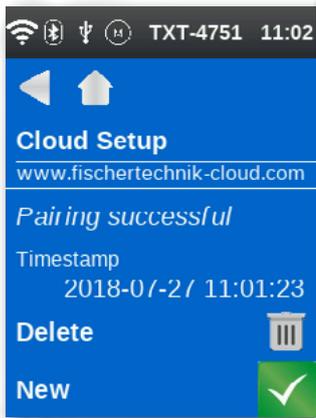


If the TXT Controller is able to establish a connection to the cloud, a QR code and a pairing code appear. You then have 30 minutes to add your TXT Controller to your account on the cloud. After expiration of this time, you will have to restart the pairing procedure.

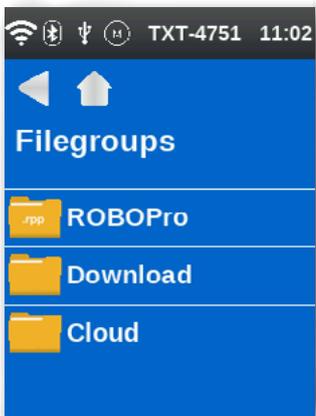
You can scan the QR code, e.g. with the "Quick Scan" app, which will then route you to the fischertechnik cloud automatically.



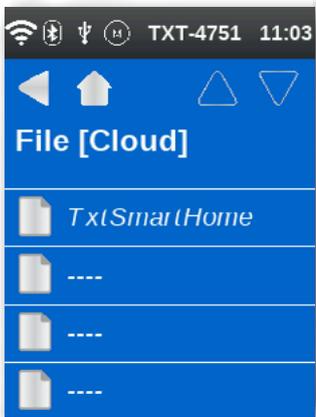
As an alternative, you can go to “Settings / Add Controller” on the fischertechnik cloud site and enter the pairing code manually.



Your TXT Controller is now connected to the cloud.



Select the application TXTSmartHome under ...



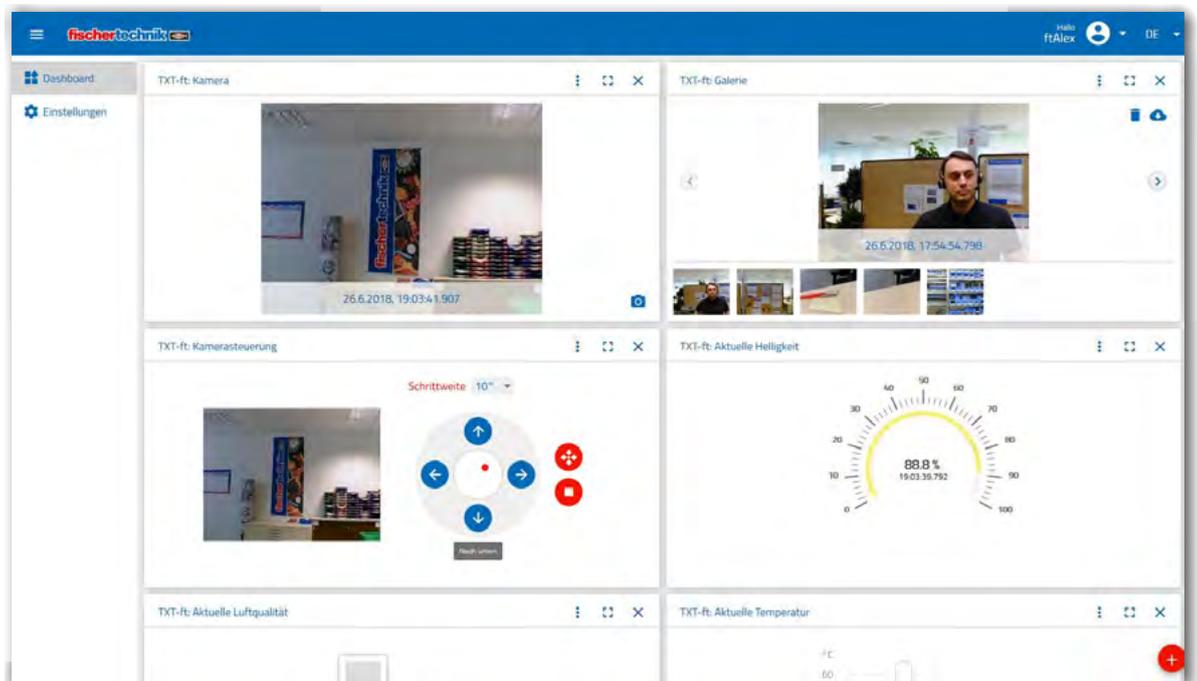
... “File - Cloud” on your TXT Controller and start it.

Now you can control the Sensor Station model via the cloud, almost like the previous ROBO Pro programs.

What advantages does this type of programming offer? You are no longer limited geographically; you can control the model from anywhere around the globe with your Smartphone, tablet or any PC with access to the fischertechnik cloud via the Internet.

You can display the sensor readings on the so-called “Dashboard”, see the image on the camera and control the camera with the direction buttons.

Now you always know what is going on in the area around your sensor station.



The figure shows the current dashboard with the camera image, camera control, a picture gallery and the temperature reading.

The following features are available on the fischertechnik cloud:

“Display window on dashboard for temperature, humidity, atmospheric pressure, air quality, brightness, camera image, camera control and gallery for snapshots (made with the camera).

The individual images can be added with the red + button at the bottom right in the window.



The menu in each window allows you to switch the sensor readings over between display of a single value and depiction as a curve.



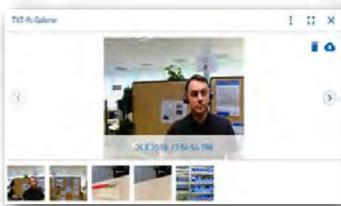
The sensor readings can be exported in a *.csv file.



You can move the individual windows as desired and change their display sequence. Each window can be maximized for viewing in the full display mode.

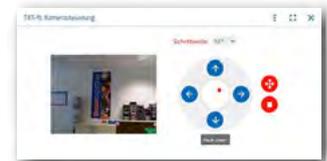


Individual images can be recorded with the camera.



The images in the gallery can be downloaded individually or all together. Images can also be deleted.

Camera control: You can control the two camera axes separately. It is possible to adjust the increment width. The red dot in the middle of the camera control shows the camera's position within the available pan range.



Under profile (menu at top right) you can change your password as well as delete the entire profile.

A number of TXT Controllers can be assigned to each profile.

For each TXT Controller it is possible to set whether the TXT Controller is to send an email or a browser message for the following alarms: (Motion in camera image, Temperature less than 4°C (frost monitor), Humidity greater than 80% (mold monitor)).

The desired language can be set in the dashboard menu.

We wish you a great deal of enjoyment working out the individual program features.

Your fischertechnik team

Trouble Shooting

Interface Test

Once again here our advice: Check each individual component for proper function with the aid of the ROBO Pro interface tests.

If things don't work right from the very start there is usually a simple reason. But, it is not always easy to find. That's why we want to give you some information here about possible sources of errors.

Cables and Wiring

If an electrical component does not work at all, check the cable used to connect it to the ROBOTICS TXT Controller again.

Other error sources are incorrectly installed plugs (for example, a green plug on a red cable).

Also check that "+" and "-" are correctly connected. For this purpose, compare your model with the illustrations in the assembly instructions.

Loose (Intermittent) Connections

A component, that works sometimes and doesn't work at other times, probably has a loose connection somewhere in the wiring for the component.

The most frequent causes for this are:

- **Loose Plugs**

When the plugs for the cables are too loose, that is wobbly in the jack sockets, they do not provide sufficient contact. In this case you can carefully bend the contact springs apart at the front on the affected connectors using a screwdriver. Just a little, so the plugs are held firmly in the jack sockets when you plug them in.

- **Poor Contact between Cable and Plug**

Also check the contact between the stripped ends of the cable in the connector and the connector itself. It may be sufficient to tighten the screws in the connector slightly.

Short Circuits

When a positive and negative connection touch one another you have a short circuit. The battery as well as the [ROBOTICS TXT Controller](#) have a built-in fuse to keep them from being damaged by a short circuit. They simply switch off the power supply for a while. Naturally, your model will not work anymore either.

The cause for a short circuit can be either a mistake in the wiring or screws which have not been tightened sufficiently in the connectors. They can touch when the connectors are plugged in, causing a short circuit. That's why you should always completely screw in the screws and plug in the plugs so that the screws cannot touch each other.

Errors in the Program

Even if no one likes to admit it: Everyone makes mistakes. And especially with complex programs, an error can creep in quite quickly.

If your model still doesn't do what you want it to after you have checked everything on the model itself and have remedied any faults, you should also check your program. Go through it line for line to see if you can find the error.

You can also watch the program running on the monitor in the online mode, which means with the [ROBOTICS TXT Controller](#) connected to the PC. The program element active at any given moment is highlighted allowing you to always see the point where the program is and where the error occurs.

Last Sources for Help

If, in spite of all this, you have not found the error, there are still two possibilities for obtaining help:

- **Email Help**

Send an email to fischertechnik describing the problem.

Our email address is: info@fischertechnik.de.

- **Internet Help**

You can also visit our web site on the Internet at

<http://www.fischertechnik.de>. Our Service FAQ's on this site offer very helpful answers to numerous questions. In addition, you can also become a member of the fischertechnik Fan Club at no charge.