# Smart Robots Pro

## Getting Started

For your first steps in the world of Smart Robots Pro we recommend that you watch the introductory video. Among other things, you will learn how to connect up with the controller, how to do an interface test and carry out the controller configuration.

Then you can start programming.

It is worth building and programming the models in the given order. You gradually get to know new programming modules that you need in later tasks.

Have fun!
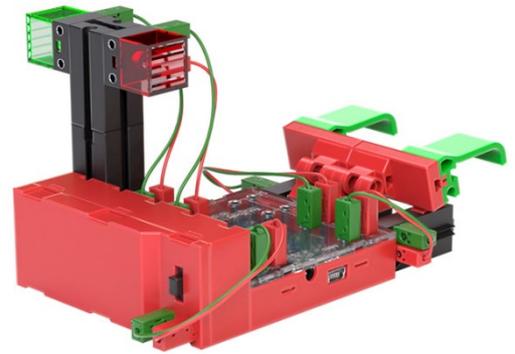
## Carousel

Last week, Svenja was at the fair and was able to test lots of different fairground rides. She liked the carousel best. She couldn't get enough of it. That's why Svenja decided to build her own carousel. Can you help her with assembly and programming?

- Build the model using the building instructions.

- Connect the cables according to the circuit diagram.

- Software ROBO Pro Coding software.

- Connect the BT-Smart Controller with the PC or another mobile device via the Bluetooth interface.
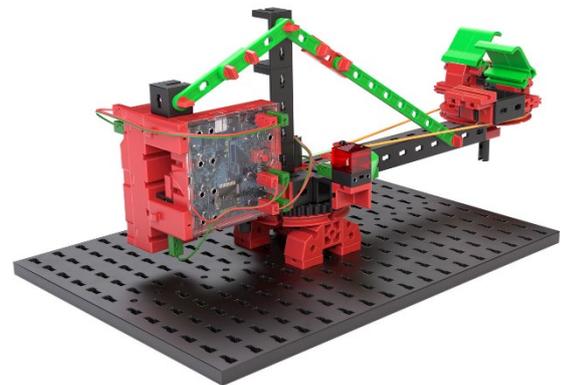
Task 1:

When the controller is connected, you can carry out the first test run using the interface test.

To do this, simply slide the controller of motor 1 to the right and observe how the carousel rotates. You can make the carousel move in the other direction as well by changing the direction of rotation to left.

*Tip: If you are not sure where the interface test is and how it works, watch the introductory video again.*

The carousel has built-in lights, you can switch these on and off in the interface test using the controller for motor 2. What happens when you change the "direction" of the second output (M2)?

Task 2:

Svenja has already found a template for a carousel program on the internet. But she is not quite sure what the program does. Can you recognise what the effect of the workflow shown is on your carousel?
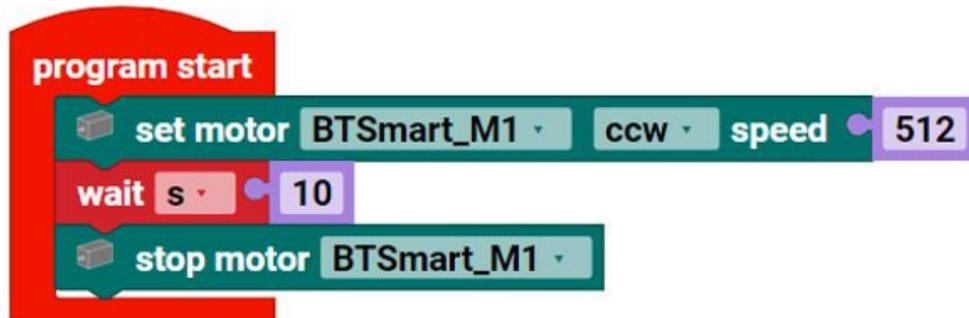


Test the program and see what happens.
The controller configuration must be carried out in ROBO Pro Coding for this. If you don't know where to find this, have another look at the introductory video.

Solution: The motor is started, runs for 10 seconds and is then stopped again.

Task 3:

Svenja is worried that passengers could bump their heads due to the carousel starting quickly. So she has an idea: The carousel should only run at half speed (256) for the first 3 seconds and then accelerate to maximum speed (512). Can you help her to write such a program?

Task 4:

To be better able to recognise whether the carousel is rotating at slow or fast speed, the green LED is to light up at slow speed and the red LED light up at fast speed. Can you add this function to your current program?

*Tip: LEDs only work in one "direction". This means the LED lights up when the current flows from + to –. But it does not light up when the current flows from – to +. We can make use of this property, so that we can actuate two lamps with one output. To do this, we change the LED to a motor in the controller configuration, because with this we can change the direction of rotation, in other words also the direction in which current flows. If the motor input is set to anti-clockwise rotation, the green LED lights up, and if it is set to clockwise rotation, the red LED lights up. Try this function out in the interface test first. Then you can use it in your program.*

Task 5 (advanced):

Before the carousel starts moving, the red LED should flash 3x. After the fast phase, the speed is halved again for three seconds. Then the direction of rotation of the carousel changes, also with a 3-second slow and 7-second fast phase. The ride ends with another 3-second slow phase.
The green LED lights up at slow speed the red LED lights up at fast speed.

*Tip: During programming, concatenated commands are often required several times in succession. To avoid having to program an infinite number of blocks, you can use so-called loops. Here you can see two programs that have the same effect, that is that the output M1 is switched on and off (or flashes) 4x in succession.*
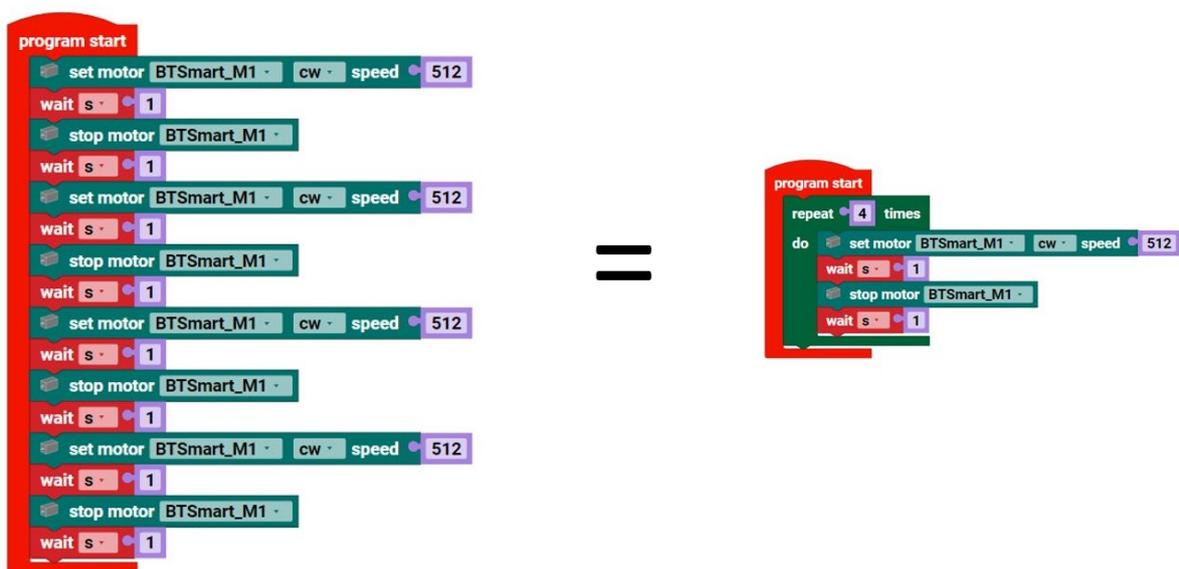


You will find example solutions for the model in ROBO Pro Coding.

## Useless Machine

Nik's little brother Ole sometimes pinches Nik's sweets. So Nik decides to build a little box that only opens for a very short time. Since Ole is not so quick, Nik can take the sweets out of the box, but Ole can't.
In addition, it is fun to operate and observe the box.
Can you help Nik to build and program such a box?

- Build the model using the building instructions.

- Connect the cables according to the circuit diagram.

- Software ROBO Pro Coding software.

- Connect the BT-Smart Controller with the PC or another mobile device via the Bluetooth or USB interface.

- Check whether all electronic components are connected correctly with the aid of the interface test.

- Then carry out the controller configuration.

Task 1:

You can move all motors to their initial position using the interface test. Check whether the pushbuttons are opened or closed.
Try to open the snout of the box via the interface test, make the actuation lever extend and retract and then close the snout again.

*Tip: The actuation lever might not switch the switch off immediately with the force of the motor. In this case, actuate the switch with the black rotary joint 10x by hand right to the front and right to the back and check whether your battery is still supplying enough voltage.*
*Avoid the motors blocking for a longer period, in other words trying to rotate but not being able to because they collide.*

Task 2:

Now create a program for the Useless Machine:
When pushbutton I4 is actuated, the snout of the box should open and stop after 0.6 seconds. Then the actuation lever extends and opens pushbutton I4 again. Then the actuation lever returns to the initial position so that pushbutton I2 is closed. Then the snout closes again.

You can use the **wait until** block for this program. This stops a program until the dependent condition has been fulfilled and only then continues to run.

You will find example solutions for the model in ROBO Pro Coding.

## Dancing robot

Clara loves the carnival and parades of all kinds. But her hands always get tired from waving. So that she can still wave to all the beautiful floats in the town's parade, she simply builds a robot that waves for and with her.

- Build the model using the building instructions.

- Connect the cables according to the circuit diagram.

- Software ROBO Pro Coding software.

- Connect the BT-Smart Controller with the PC or another mobile device via the Bluetooth or USB interface.

- Check whether all electronic components are connected correctly with the aid of the interface test.

- Then carry out the controller configuration.

Task 1:

Use the interface test to move the position of the arms in such a way that the left arm is pointing vertically down.
Program the robot in such a way that it waves permanently, in that it moves its arms in a loop at speed 100 for 2 seconds, then stops for one second, then moves its arms in the other direction for two seconds. After these two seconds the arms stop again for one second.
Make sure that you move the arms back into the initial position by means of the interface test before every program start, and that the motors do not block.

You can vary the speed as you like – and can then adapt the duration as well.

Task 2:

To draw even more attention to the dancing robot, Clara wants to make the robot nod after every second wave, by rotating the cam disc anticlockwise at speed 350 for 2 seconds after every second wave. Can you help her to write a corresponding program?
Make sure that you move the arms back into the initial position by means of the interface test before every program start, and that the motors do not block.

You will find example solutions for the model in ROBO Pro Coding.

# Drivebot

- Build the model using the building instructions.

- Connect the cables according to the circuit diagram.

- Software ROBO Pro Coding software.

- Connect the BT-Smart Controller with the PC or another mobile device via the Bluetooth interface.

- Check whether all electronic components are connected correctly with the aid of the interface test.

- Then carry out the controller configuration.

Task 1:

Program the robot in such a way that it moves straight on after the program has started. If the robot meets an obstacle, in other words one of the NC switches is actuated, the robot reverses for one second and then turns, in that one wheel moves forwards for one second and the other backwards. The speed of the wheels during turning is 200 for two seconds. Then the robot darts off straight ahead again.

Pushbuttons can be used as NC or NO switches. The condition of a NO switch is 0 in the non-actuated state and 1 in the actuated state. This is the other way round with the NC switch: when non-actuated the state is 1 and actuated 0.

*Tip: Make sure that the joints of the fire button are pushed right to the top so that the pushbuttons can be actuated reliably.*

Task 2:

Adapt the program in such a way that the robot corrects is path to the left if it collides with an obstacle on the right, and, vice versa, corrects its path to the right if it collides an obstacle on the left.

You will find example solutions for the model in ROBO Pro Coding.

# Teach-in Robot

Teach-in robots can be taught what to do by means of input directly at the robot. They are complicated to program, so you can open the example program in ROBO Pro Coding directly.

- Build the model using the building instructions.

- Connect the cables according to the circuit diagram.

- Software ROBO Pro Coding software.

- Connect the BT-Smart Controller with the PC or another mobile device via the Bluetooth interface.

- Check whether all electronic components are connected correctly with the aid of the interface test.

- Then carry out the controller configuration.

Task 1:

Build an obstacle course. Then try to master this with the teach-in commands. The pushbutton input works more reliably if you actuate it for several seconds per command.

Left-hand pushbutton: Anti-clockwise rotation (90°)
Right-hand pushbutton: Clockwise rotation (90°)
Both pushbuttons at the same time: Travel straight forward

The robot is pre-programmed to start moving after 5 commands. But you can change the number by changing the value of the variable list_move < 5. In addition, you can see what happens when you change the speeds.

## Alarm system

In Germany, there were more than 65,000 burglaries in 2022. This corresponds to 178 burglaries per day. It's good if you know how to help yourself and protect yourself through technology. To get to know how an alarm system works, we are going to build one of our own.

- Build the model using the building instructions.

- Connect the cables according to the circuit diagram.

- Software ROBO Pro Coding software.

- Connect the BT-Smart Controller with the PC or another mobile device via the Bluetooth or USB interface.

- Check whether all electronic components are connected correctly with the aid of the interface test.

- Then carry out the controller configuration.

Task 1:

Program the alarm system as follows:

- If the main switch is not actuated, nothing is to happen: You and your family can open and close the door normally.

- If the main switch has been actuated and the door is closed so that the Reed contact has the status 1, the green LED lights up.

- If the main switch has been actuated but the door is open, the red LED flashes and the rattle will make a noise for one second every two seconds.

- If the door is closed again after this, the alarm should remain triggered.

- The alarm only stops again when the main switch has been switched off again.

You will find example solutions for the model in ROBO Pro Coding.

# Fan

Fans are very simple but useful machines. A simple fan can be switched on and off, some can also be pivoted from one side to the other. We now want to build an intelligent fan that starts these two functions automatically.
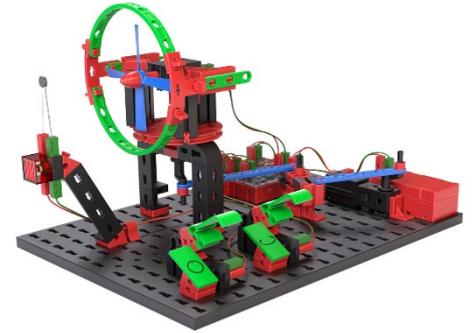
- Build the model using the building instructions.

- Connect the cables according to the circuit diagram.

- Software ROBO Pro Coding software.

- Connect the BT-Smart Controller with the PC or another mobile device via the Bluetooth or USB interface.

- Check whether all electronic components are connected correctly with the aid of the interface test.

- Then carry out the controller configuration.

Task 1:

Program the fan in such a way that it can be rotated and pivoted backwards and forwards via the two switches. The fan should only pivot backwards and forwards when it is rotating as well.

Task 2:

Make the fan control intelligent in that the fan automatically starts up as soon as the NTC resistor value is smaller than 1300. If the value is exceeded, the fan can still be controlled via the switches.

We make the controller calculate for this. You have to change to level 2 for this. Here, we compare in a logic block whether the sensor value is smaller (<) than a specified value.

Tip: You can adapt the values if necessary. Look in the interface test which value the NTC resistor outputs in the "cold state". You can simulate the heat by touching the NTC resistor with your fingers.

You will find example solutions for the model in ROBO Pro Coding.

# Component dispenser

- Build the model using the building instructions.

- Connect the cables according to the circuit diagram.

- Software ROBO Pro Coding software.

- Connect the BT-Smart Controller with the PC or another mobile device via the Bluetooth or USB interface.

- Check whether all electronic components are connected correctly with the aid of the interface test.

- Then carry out the controller configuration.
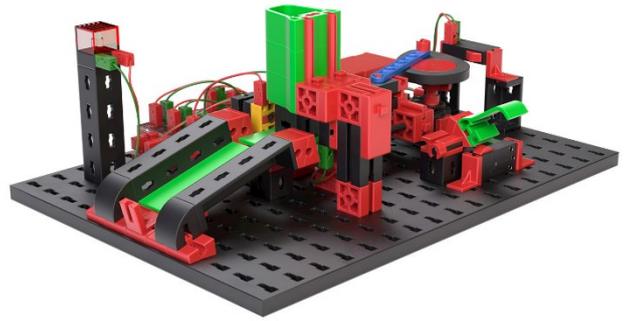
Task 1:

Program the component dispenser in such a way that it dispenses one component every time the pushbutton (I1) is pressed. If no more components are detected in the magazine, the signal LED lights up and the dispenser no longer reacts to the button being pressed.

Task 2:

Turn your component dispenser into a component counter. After the trigger has been actuated, the counter pushes component after component out of the magazine until this is empty. The components are to be counted at the same time and the result outputted as text.

You can use variables for this. You can create a variable with the name "count" and set this to 0 at the start of the program. With every component that is ejected, you can add +1 to the variable.

You will find example solutions for the model in ROBO Pro Coding.

# Drawing robot

- Build the model using the building instructions.

- Connect the cables according to the circuit diagram.

- Software ROBO Pro Coding software.

- Connect the BT-Smart Controller with the PC or another mobile device via the Bluetooth or USB interface.

- Check whether all electronic components are connected correctly with the aid of the interface test.

Task 1:

Grab a piece of paper and use two pieces of adhesive tape to stick it to the table. Take a pen that fits into the pen holder of the drawing robot and fasten it in the holder. Then align the drawing robot to the paper.

In the interface test you can now create numerous patterns by changing the motor speeds and directions of rotation. If you especially like a pattern, you can create a program with the chosen parameters so that you can have it drawn repeatedly. Make the drawing robot draw for a while until you can recognise a pattern.

Always use one piece of paper when using the drawing robot and make sure the paper does not let ink through.

Task 2:

Another factor that influences the patterns are the rotary axes. Try to see what happens when you change the motor positions.

# Ball game



- Build the model using the building instructions.

- Connect the cables according to the circuit diagram.

- Software ROBO Pro Coding.

- Connect the BT-Smart Controller with the PC or another mobile device via the Bluetooth or USB interface.

- Check whether all electronic components are connected correctly with the aid of the interface test.

- Then carry out the controller configuration.

Task 1:

Create a program that triggers a throw every time the button is pressed. Make sure the direction of rotation is correct for the throw. Interrupt the program if the direction of rotation is not correct and the motor blocks.

In the interface test you can adapt the tilt of the ejector by controlling the second motor. Choose a flat angle initially.

You can adjust the power of the throwing mechanism. If the ball does not fly far enough, you can move the strut to the left and right first. Make sure that the strut is pressed back by the rotating lever during the shot. If the lever is not strong enough to push the strut back, check the battery voltage and replace it if necessary.
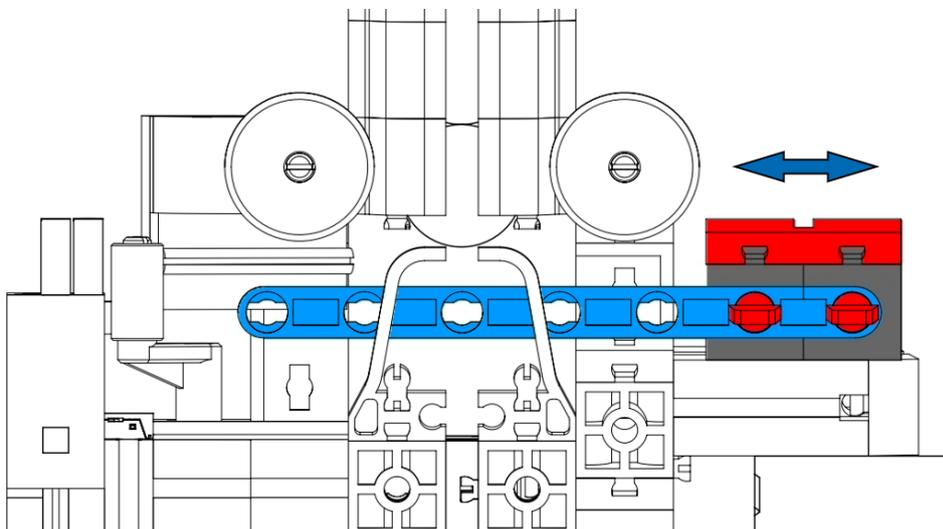
To boost the throw further, you can reinforce the strut holder using the following components:

Building block 5 | 37237
Angular block 7.5°| 32071
Angular block 10x15x15 | 38423
Spring cam | 31982

To do this, push the strut holder out of the model, take building block 5 off the holder and add the listed components. Then slide the reinforced holder back into the model.
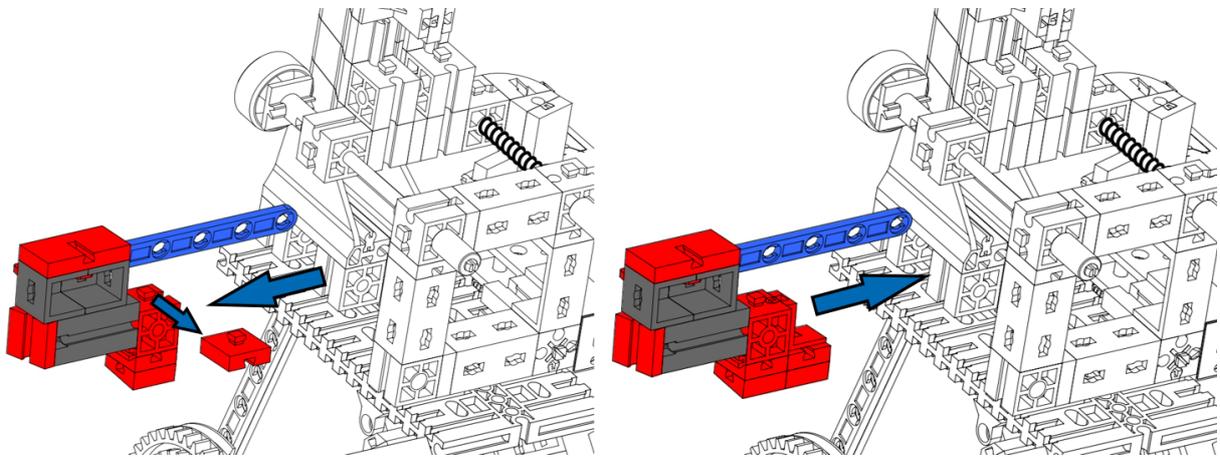


Task 2:

To make hitting the target a little harder, you can change the angle continuously. Just supplement your program in such a way that motor 2 runs continuously.

A particular challenge is when motor 2 suddenly changes its direction of rotation or speed. Can you add this enhancement to your program?

You will find example solutions for the model in ROBO Pro Coding.

# Swingboat

- Build the model using the building instructions.

- Connect the cables according to the circuit diagram.

- Software ROBO Pro Coding software.

- Connect the BT-Smart Controller with the PC or another mobile device via the Bluetooth or USB interface.

- Check whether all electronic components are connected correctly with the aid of the interface test.

- Then carry out the controller configuration.

While building the model, make sure that the boat is just about in contact with the tyre. Then pivot the swing by hand and use the interface test to make sure that the Reed contact trips when the magnet on the swingboat passes the Reed contact.

Task 1:

Create a program that makes the swingboat speed up due to the tyre rotating. As soon as the Reed contact is closed, the tyre should rotate for approx. 0.2 seconds on the other direction. Then the direction of rotation changes again until the Reed contact is closed again.
The program should not start until the main switch (I1) on the model is closed. As soon as the main switch is opened again, the swingboat should swing outwards again, actuate the Reed contact 1x, swing back in the other direction and then stop, so that the boat is vertical at the tyre.

Tip: For the tyre to accelerate the swingboat reliably, you can slide it to the left or right using the motor. The tyre should always touch the swingboat when it has swung out.

Task 2:

Create a variable with the name "rounds" and save the value 4 there. Before the ride starts, the LEDs should flash alternately, as often as the size of the rounds value. Then the swingboat swings as often as the size of the rounds value.

Can you manage to assign the variable a random value between 1 and 6?

You will find example solutions for the model in ROBO Pro Coding.