2016

University of Jordan
Faculty of Engineering and Technology
Mechatronics Engineering Department

Control and Measurement Laboratory

Robotino Robot
(Mobile Robot System)

Robotino Robot

Objectives:

The following are special characteristics of and special requirements for all mobile robots:

- Mobile machines with autonomous orientation, navigation, obstacle recognition and avoidance
- Autonomous power and computer supply
- Incorporation of own sensors and actuators

The Robotino® learning system meets all these requirements and enables you to familiarize yourself with the multifaceted technical areas of knowledge of mobile robotics.

Following are the objectives of the experiment:

- Commissioning of a mechatronic system
- Acquisition and scaling of miscellaneous sensor data
- Electrical motor control/drive unit
- Electrical drive technology
- Closed-loop control of a mechatronic system
- Graphic programming of applications for a mobile robot system
- Analysis of sensor data for various applications

Introduction:

A mobile robot is an automatic machine that is capable of movement in a given environment.

Mobile robots have the capability to move around in their environment and are not fixed to one physical location. In contrast, industrial robots usually consist of a jointed arm (multi-linked manipulator) and gripper assembly (or end effector) that is attached to a fixed surface.

Mobile robots may be classified by:

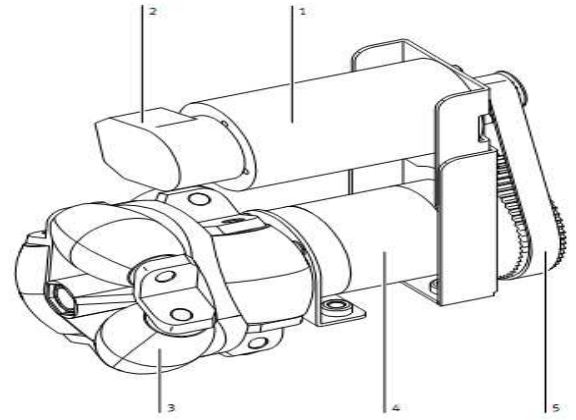
- The environment in which they travel:
  - Land or home robots. They are most commonly wheeled, but also include legged robots with two or more legs (humanoid, or resembling animals or insects).
  - Aerial robots are usually referred to as unmanned aerial vehicles (UAVs)
  - Underwater robots are usually called autonomous underwater vehicles (AUVs)
  - Polar robots, designed to navigate icy, crevasse filled environments

- The device they use to move, mainly:
  - Legged robot : human-like legs (i.e. an android) or animal-like legs.
  - Wheeled robot.
  - Tracks.

Robotino is a fully functional, high quality mobile robot system with omnidirectional drive. The three drive units allow for motion in all directions – forward, backward and sideways – and the robot can be turned on the spot as well. It is also equipped with a webcam and several types of sensors, analogous to distance measurement, for example binary for collision protection and digital to check the actual speed. This assures that all of the wide ranging demands placed upon systems of this type are fulfilled. The system can be placed into service immediately – without a PC.

Drive system:

- 3 drive units consists of the following components
- DC motor
- Gear unit with a gear ratio of 16:1
- All-way roller
- Toothed belt
- Incremental encoder



1.Motor 2.Incremental encoder 3.All way roller
4.Gear unit 5.Toothed belt

Sensors:

- 9 Infrared distance measuring sensors
- 3 Incremental encoder
- The anti-collision sensor (bumper)

1.Incremental encoder 2.Impact, anti-collision sensors 3.Distance measuring sensors

Preliminary to Assignments

A. Assignment 1: Linear travelling of a mobile robot system in any direction

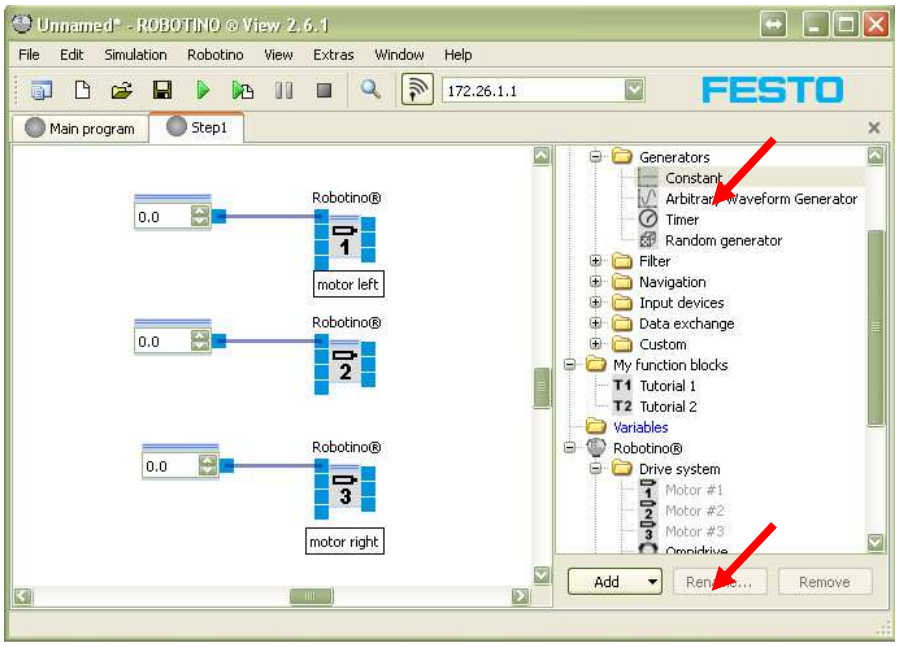
Actuation of motor, forward movement of robot system

1-Open Robotino view from program menu.

2-Open a blank function block diagram in Robotino View as you see the screen is separated to two areas the first one on the left hand is the programming area the right hand one contains blocks for the the robot drive unit, sensors motors and for different logic and mathematic operations.

3-Drag the constant block from generators as shown in the picture.

4-Drag the three motors and then connect the blocks as shown in the following figure.



Now move the robot in the forward direction what do you notice about the three motors speed and direction?

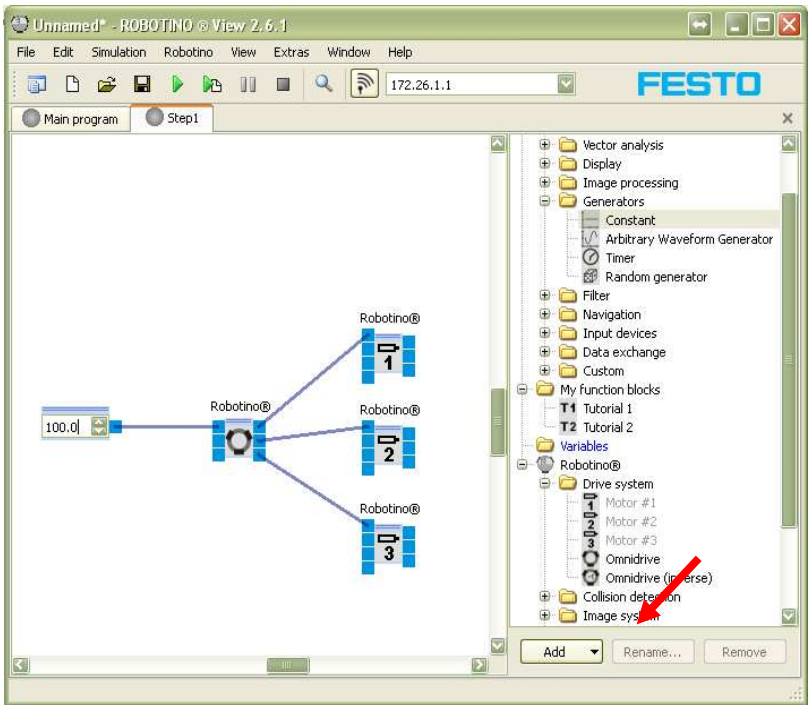
What is the constants suitable values that moves the robot in the forward direction. Note that the speed is in rpm and the direction is indicated by negative sign.

Let the speed be 100 rpm now set the suitable values for the constants block and then start connection by pressing on the button shown in the following figure. sure that the IP address for the Robotino is correct.
When it converts to green instead of gray that means the connection is establish now run the program by pressing on start main program button avoid collision and press stop when needed

B. Assignment 2: Omnidrive

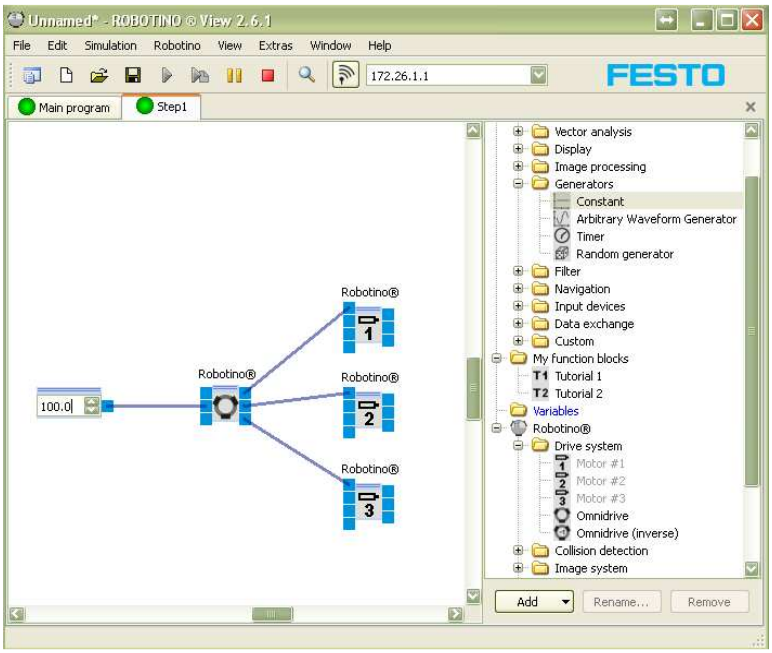
1-Open a new blank function block diagram in robotino view.

2-drag the three motors and omnidrive block(from robotino-drive system) and constant from generators and connect the blocks as in the following figure

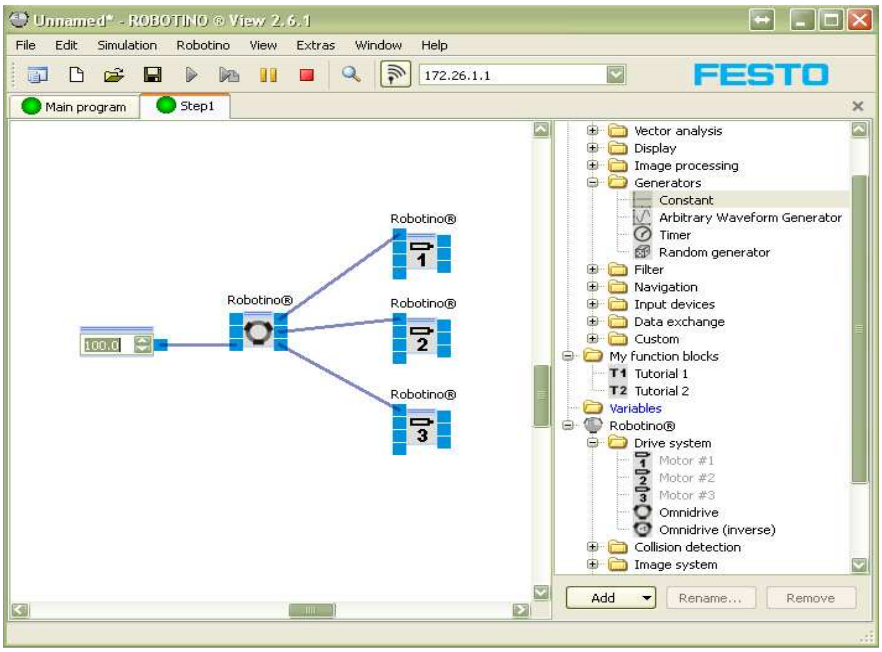


Right 100 in the constant block this means 100 mm/s. run the program and notice the motion.
Change the constant value to -100 mm/s and notice the motion again.

Now connect between the blocks as in the following figure:



Run the program, notice the motion then change the constant value to -100 and run the program again.
Again connect the blocks as follows:



Run the program and notice the motion, change the value to -100 and notice the motion again.

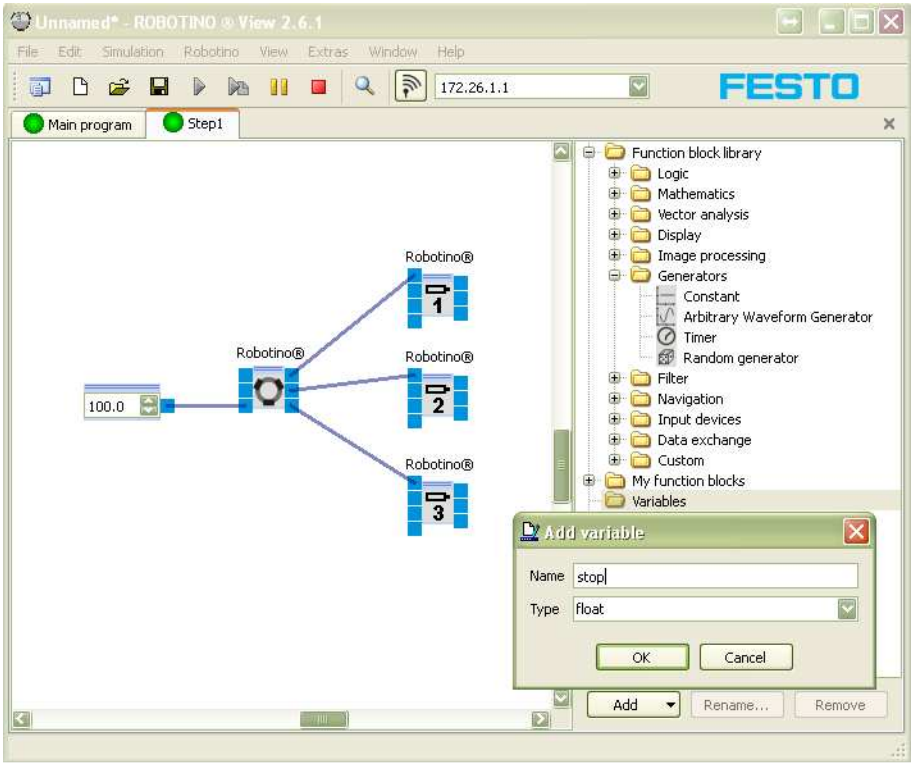
Experiment Instructions:

Exercise 1: Now let the robot moves in 45o with speed 100mm/s.

Collision protection

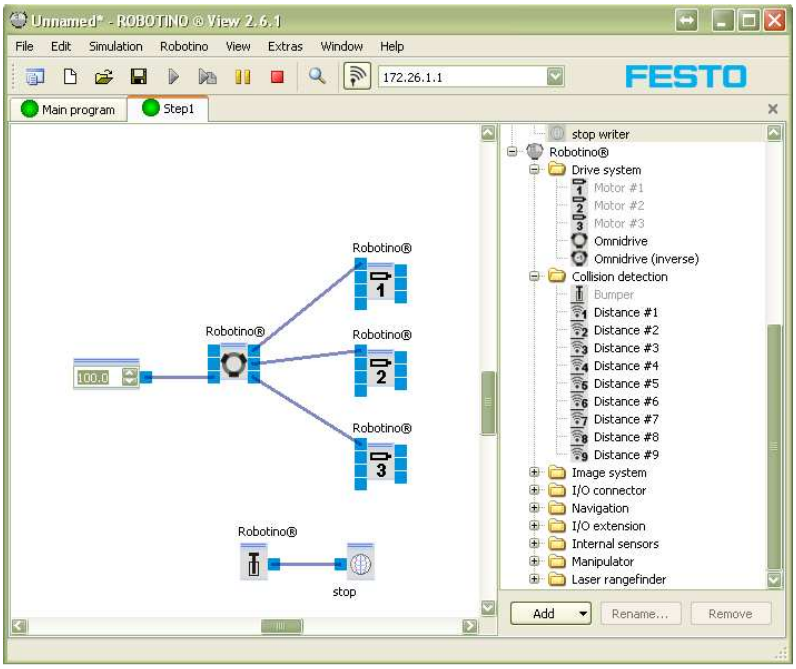
Now you are going to move the robotino in the forward direction and add collision protection to the program.

1-open the forward program, on the right menu click right on the variables the following window will appear .

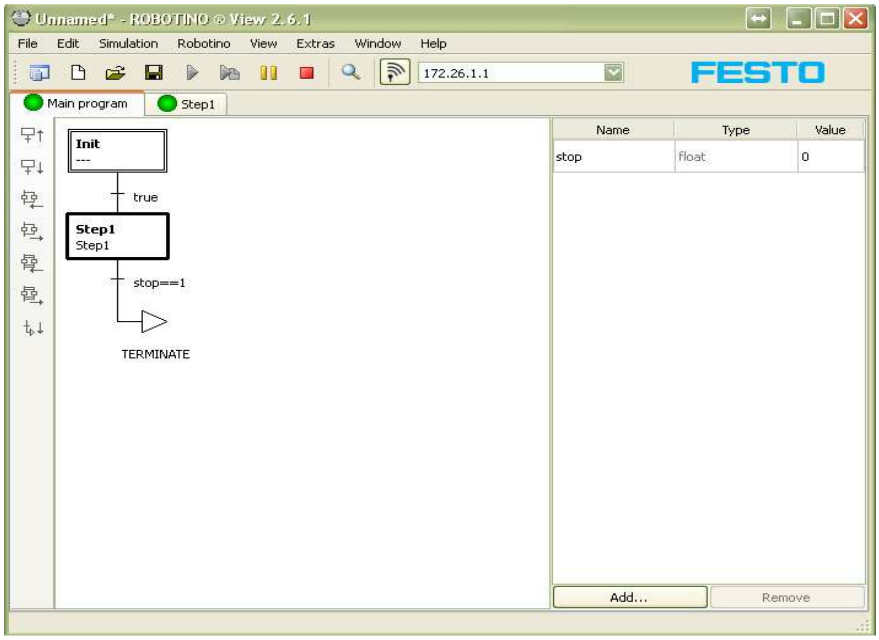


Write stop for the name then click ok.
Two blocks will appear stop reader and stop writer, drag the stop writer block to the blocks function window.

Drag bumper block from robotino collision detection menu and connect between the blocks as in the following picture.



Now open the main program on the left of the function block window change false to stop==1and initial to TERMINATE.



Run the program and notice what happens.

Accurately positioned approach of a loading station.

In order to respond to the individual distance sensors, you need to determine which sensor is mounted at any position of the Robotino

Procedure:

The position of a sensor is determined by visualizing its output value. The date display must be switched on in RobotinoView (Ctrl-D). An obstacle is placed in front of the sensor. If the output value of the sensor changes, the sensor position is determined and its function tested.

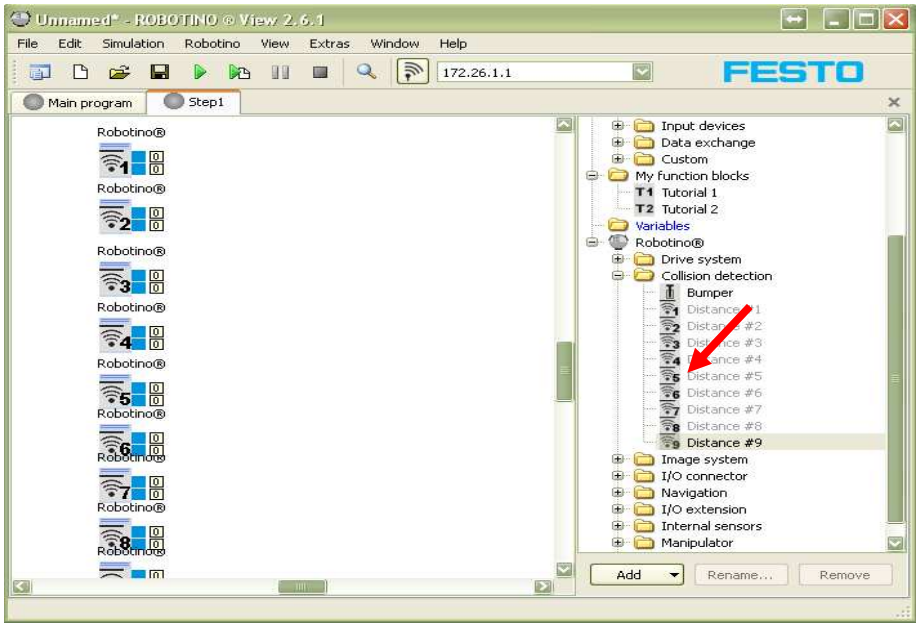
- Create a program in RobotinoView to determine the position of the distance sensors “distance 1” to “distance 9"and save the program.

- Jacked up the Robotino and switch it on.

- Open a blank function block diagram in Robotino View.

- Drag a distance sensor onto the function block diagram.

- Assign one of the distance sensors to the function block by opening the function block dialogue via a double click onto the distance sensor symbol from view menue click –show connector values.



Exercise 2:

The Robotino is to stop at a predetermined distance from the loading station. Consider how this can be achieved.

1. First determine which distance sensors are required for the distance measurement and make a note of this.
2. "Distance 1" sensor is required if you take into account the fact that the Robotino is to face the loading station. The other sensors are not required for this exercise.
3. How do you need to proceed to detect the required distance from the loading station in an open-loop control program?

For different distances, the respective sensor value needs to be determined. A characteristic curve of the distance sensor must be recorded.

Procedure:

Create and start a program with a distance sensor. Switch on the display of values in Robotino View. Place an obstacle directly in front of the selected distance sensor (1). Distance this obstacle from the sensor in steps of 1 cm and record the output values of the sensors. Transfer the pairs of values determined to a coordinate system and link these with a curve.

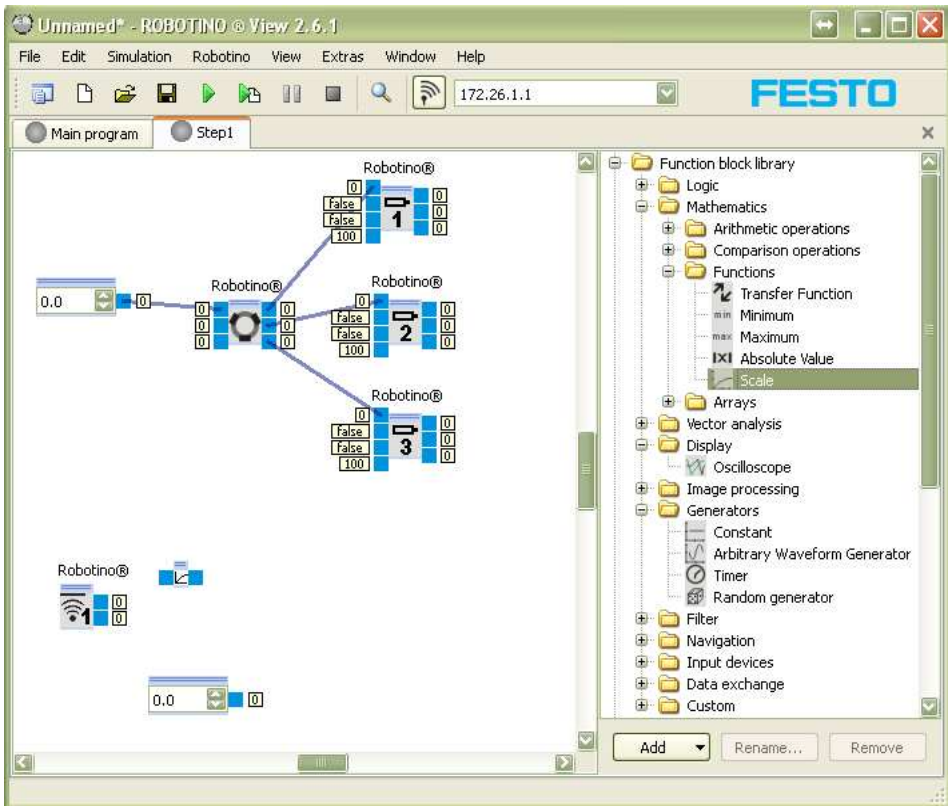
Distance (cm)	Voltage(V)	Distance (cm)	Voltage(V)
1		21	
2		22	
3		23	
4		24	
5		25	
6		26	
7		27	
8		28	
9		29	
10		30	
11		31	
12		32	
13		33	
14		34	
15		35	
16		36	
17		37	
18		38	
19		39	
20		40	

Represent the characteristics in the form of a curve using MS-Excel. Enter the coordinate of the determined pairs of values in an Excel table and represent these in the form of a diagram.

Determine the characteristic curve area required for the exercise given. Linearise the characteristic curve within this area and enter the linearised characteristic curve in your drawing or represent it in an Excel diagram. The distance should be expressed in cm.

Document the calculation in all the individual steps on the worksheet

Now drag the constant block sensor 1 and scale block from math-function as follows.



Exercise 3:

Now complete the program so that robotino moves in the forward direction then it stops when sensor 1 detects objects 6 cm far and remains 6 cm far even if you decrease the distance (use arithmetic operations).

Complete your program so the robotino moves in the forward direction till it reach a wall and follow the it (remember the closed loop control).