

LAWRENCE TECHNOLOGICAL UNIVERSITY
ROBOFEST 2022 **ROBOParade**

RoboParade

Vex IQ with Robot Mesh Parade Workshop

This file can be found under the **eAcademy > Workshops** page on the website

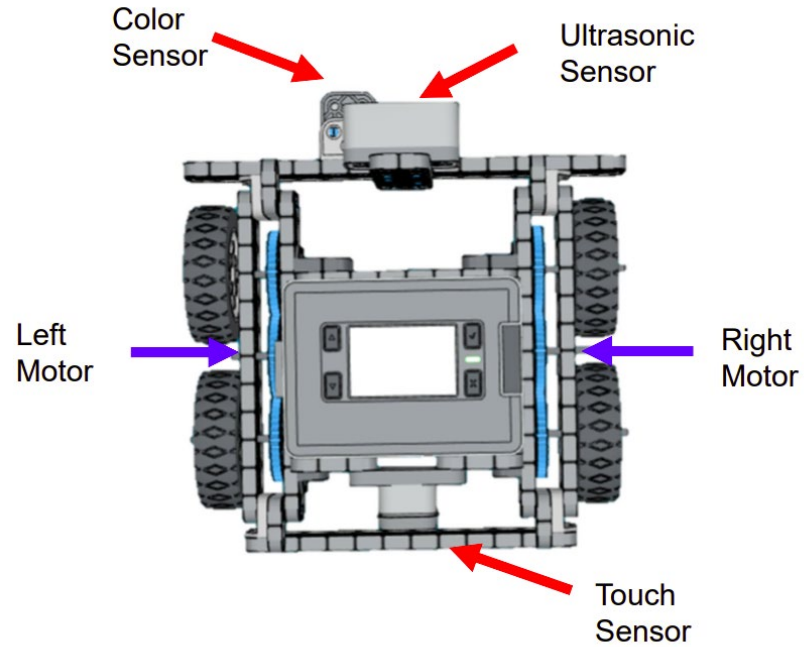
www.robofest.net

robofest@ltu.edu

248-204-3568

Room J233 Taubman Complex, LTU
21000 West 10 Mile Road, Southfield, MI 48075, USA

Vex IQ Robot Used



What is Robot Mesh Studio?

- Programming software for VexIQ
- Uses a drag and drop interface (Blockly, same as Scratch)
- Based on Python programming language



Getting Started

- Go to robotmesh.com/studio
- Go to signup and create an account
- Click on “install RMC plugin” and follow instructions to install

Install RMC plugin...

Log Out



Robot Mesh Connect Installation

Robot Mesh Connect plugin is installed correctly.

Create a New Program

- Click on “New Project”
- Make sure these are selected
 1. Target: “Vex IQ”
 2. Language: “Blocky”
- Name the program “Workshop”
- Click on “Create”

Robot MESH STUDIO

Create a New Project

New Project Name:
workshop

VEX IQ
Target

Blockly
Language

Options
Options

Click to the left to adjust project target/language.

Create

Connect Robot

- Connect one end of programming cord into computer
- Connect the other end to the robot brain
- Turn on the Robot by pressing the check button

Check button

Programming
cord



Check Connection

1. Click on
“Test Drive”

The screenshot shows the Robot Mesh Studio interface. At the top, there are navigation buttons for 'Run', 'Stop', and 'Full Speed'. Below that, there are tabs for 'Description', 'Blockly', and 'Generated Code'. The main workspace contains a 'Wheel' component. On the right side, there is a 'VEX IQ' panel with a 'Test Drive' button highlighted in red. Below the 'Test Drive' button, there is a list of components with their status: 'motor_1 Off', 'color_2 Wired', 'motor_3 Off', 'motor_4 Off', 'color_5 Wired', 'motor_6 Off', 'touch_pad_0 Off', and 'motor_11 Off'. At the bottom of the interface, there is a green bar with the text 'Connected'.

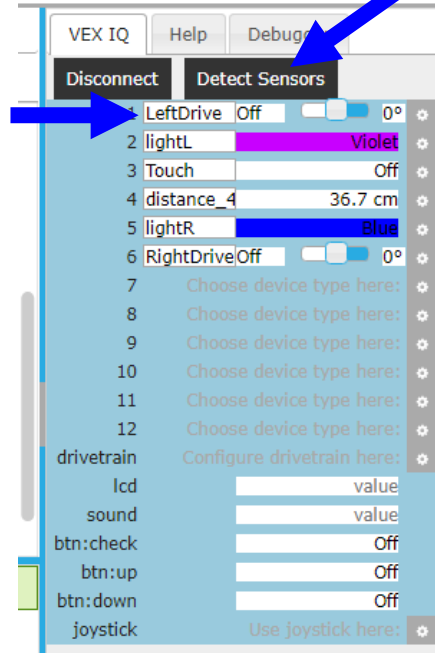
2. Look for “Connected”
message at the bottom

Copyright © 2022 Robofest

Motor and Sensor Setup

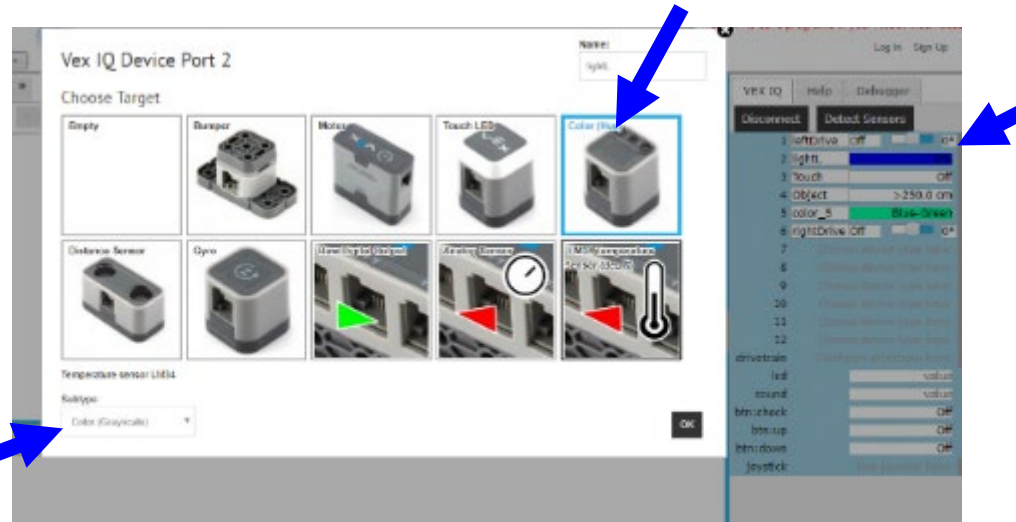
1. Click Here

2. Change to names that have meaning



Motor and Sensor Setup

Change
Color
Sensors to
grayscale
mode

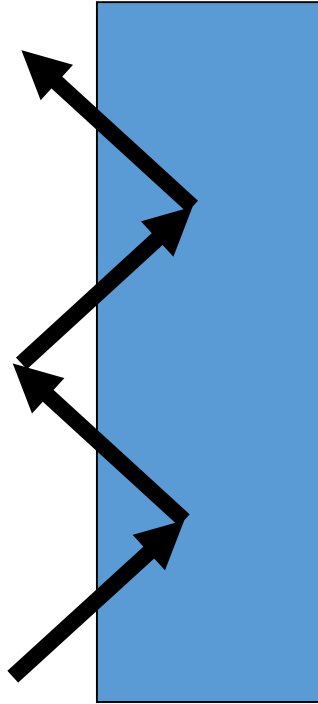


Task 1

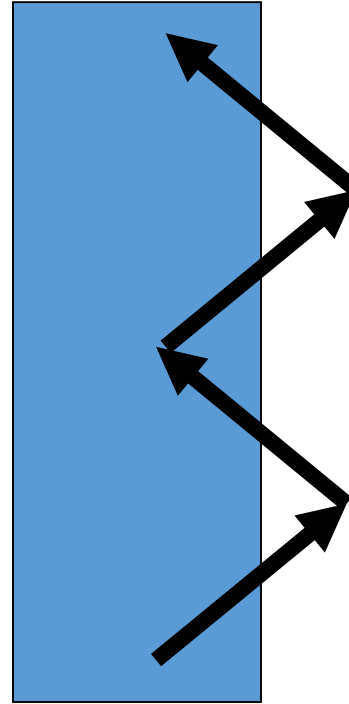
Following a Line

Zigzag Line following Idea

Left side following



Right side following



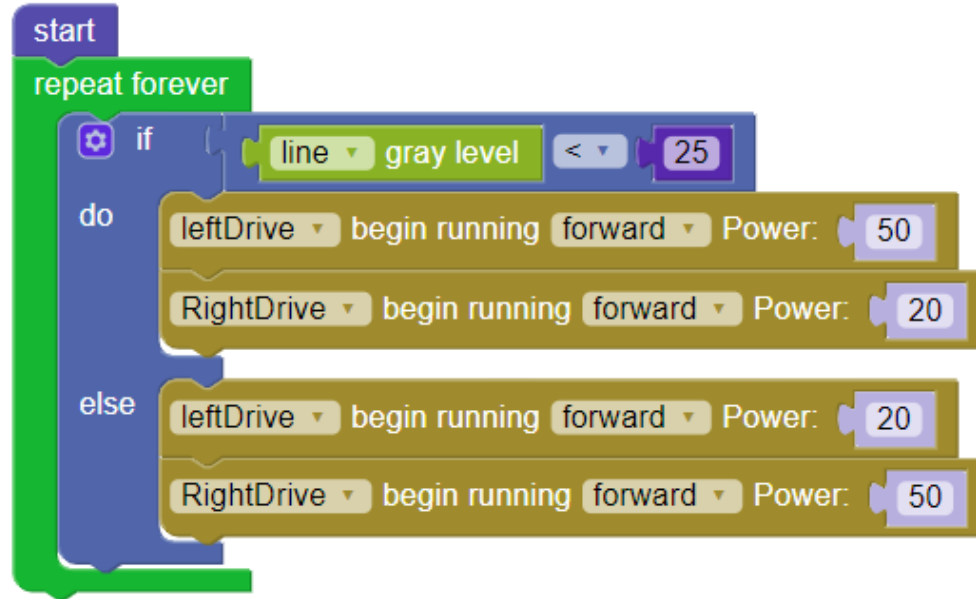
Follow a Line

- Light sensor settings example
 - Off line = 5
 - On line = 45
 - Median threshold = $(5+45)/2 = 25$

- Two cases
 - Light sensor reading > 25 . On line.
 - Light sensor reading < 25 . Off line.

Simple Line Following Algorithm

Follows right edge of line

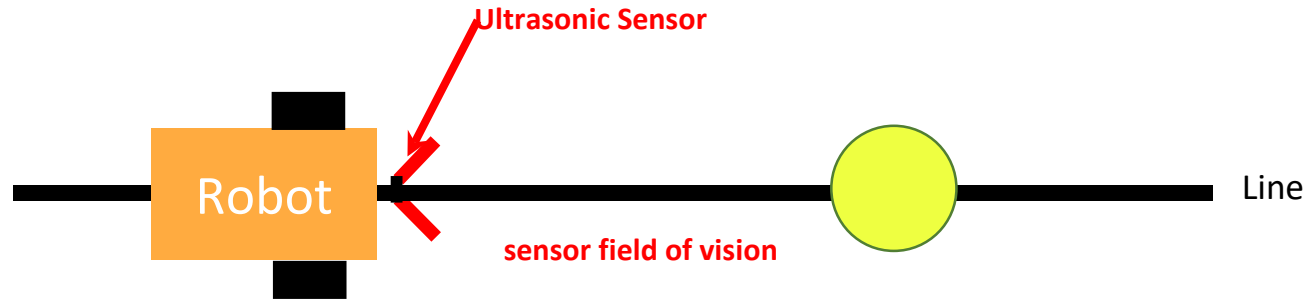


Task 2

Finding an Object

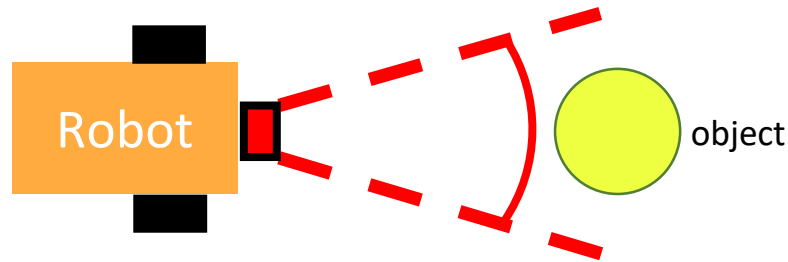
Finding a objects

- We can use a Ultrasonic sensor to determine if an object is near the robot
- Here we will assume that we are following line and wish to stop the robot once a object close to the robot



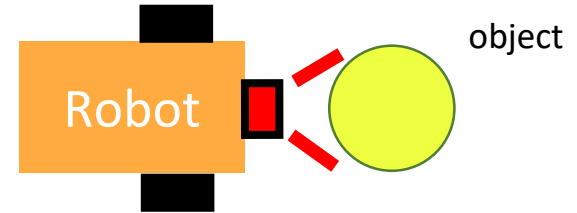
Finding a object

- Here we will use our line following program to follow line and stop the robot when the object is close r



Ultrasonic sensor will read no values when the object is far away

Starting Position

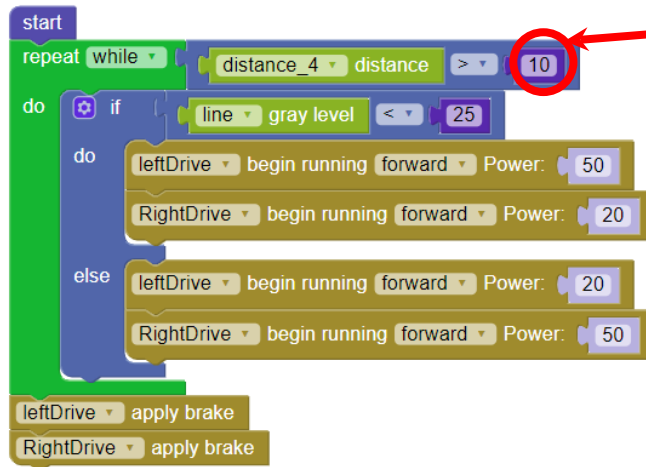


Ultrasonic sensor will read values when the object is close to the robot

Final Position

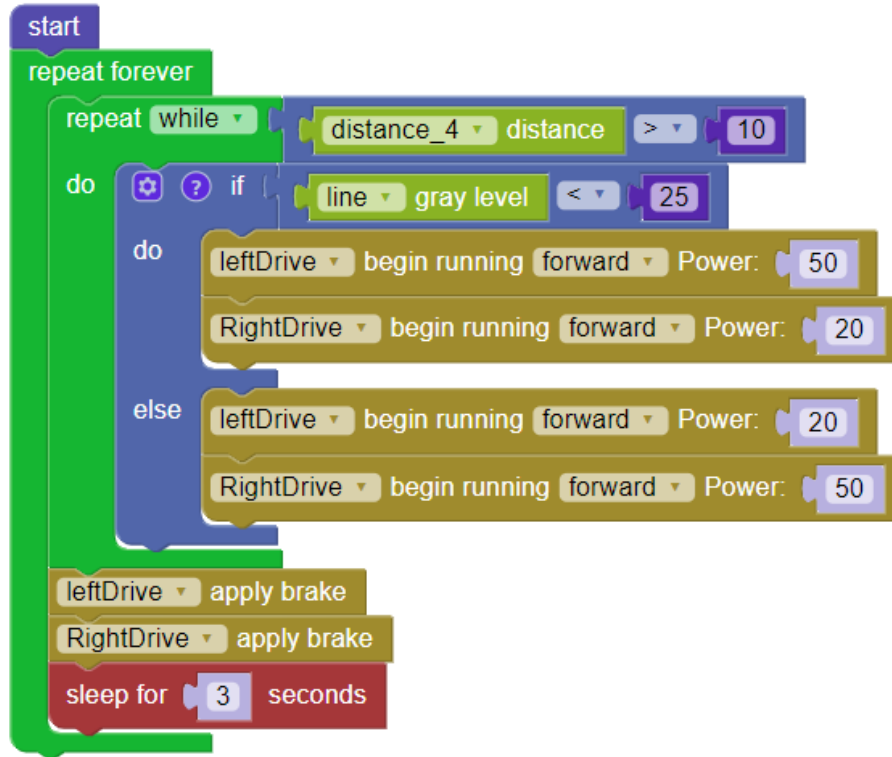
Finding an Object

- Now, we travel along the line and stop if we find a object



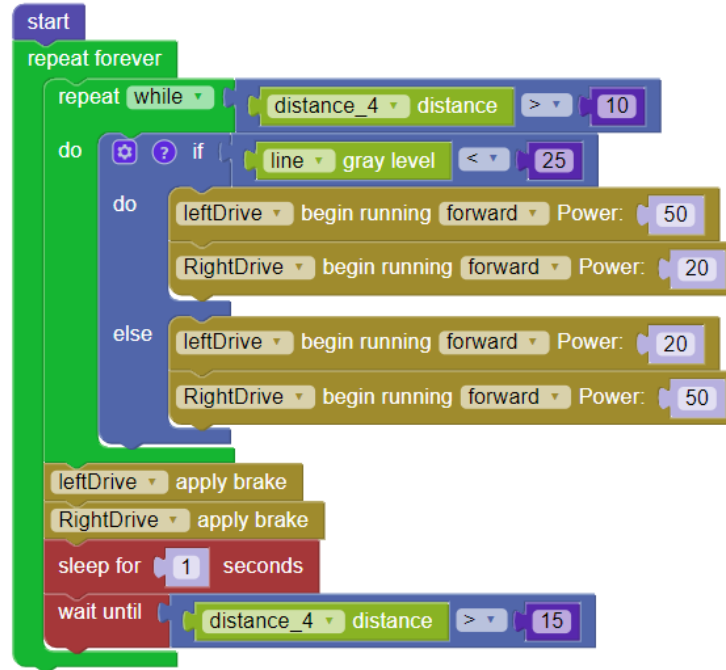
We can determine the appropriate value by testing the sensor readings with an object the front of the robot.

Finding an Object and Restart after Pause



Another Method

- Wait until clear



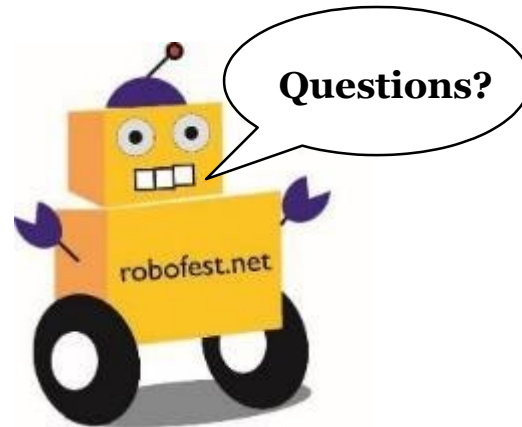
Add Additional Wait to Allow More Gap

```
start
repeat forever
  repeat while distance_4 distance > 10
  do
    if line gray level < 25
    do
      leftDrive begin running forward Power: 50
      RightDrive begin running forward Power: 20
    else
      leftDrive begin running forward Power: 20
      RightDrive begin running forward Power: 50
  leftDrive apply brake
  RightDrive apply brake
  sleep for 1 seconds
  wait until distance_4 distance > 15
  sleep for 4 seconds
```

The code is a Scratch-style block-based program. It starts with a 'start' block, followed by a 'repeat forever' loop. Inside the loop, there is a 'repeat while' block with the condition 'distance_4 distance > 10'. Within this 'repeat while' block, there is an 'if' block with the condition 'line gray level < 25'. The 'if' block has two 'do' branches. The first branch contains two 'begin running' blocks: 'leftDrive begin running forward Power: 50' and 'RightDrive begin running forward Power: 20'. The second branch contains two 'begin running' blocks: 'leftDrive begin running forward Power: 20' and 'RightDrive begin running forward Power: 50'. After the 'if' block, there are two 'apply brake' blocks: 'leftDrive apply brake' and 'RightDrive apply brake'. This is followed by a 'sleep for 1 seconds' block, a 'wait until' block with the condition 'distance_4 distance > 15', and finally a 'sleep for 4 seconds' block. The 'repeat while' block and its contents are nested within the 'repeat forever' loop.



Little Robots, Big Missions



robofest@LTU.edu
LTU Computer Science

Copyright © 2022 Robofest