



# Tutorials



## “Use the Force” with The Creative Toolbox

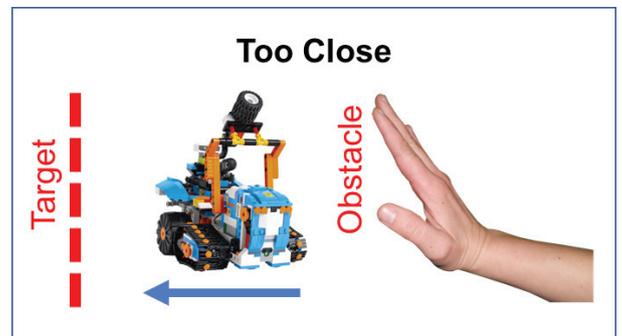
By Sanjay Seshan and Arvind Seshan

### About the Authors:

Sanjay and Arvind, or the Seshan Brothers, are the founders of [EV3Lessons.com](http://EV3Lessons.com), a popular site for learning to program with MINDSTORMS. They enjoy teaching programming and sharing their creations with others.

### Introduction:

There is a huge range of capabilities available to Boost users. You can start with the built-in models and programs and then explore further. Boost even lets you learn many advanced programming concepts.



### Lesson Objective:

In this lesson, you will use Boost Creative Toolbox to program your robot to stay at a target distance from an obstacle. The robot will back away from the obstacle if it is too close and move forward if the obstacle is further away.

We will be implementing this movement with proportional control, which makes the robot move slower or faster based on how close it is to the target distance. The pseudocode for every proportional control program consists of two stages:

1. Computing an error → how far is the robot from a target?
2. Making a correction → make the robot take an action that is proportional to the error (this is why it is called proportional control). You must multiply the error by a scaling factor to determine the correction.

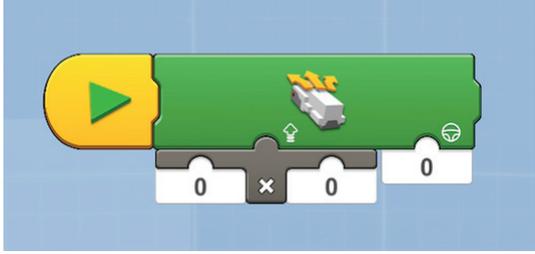
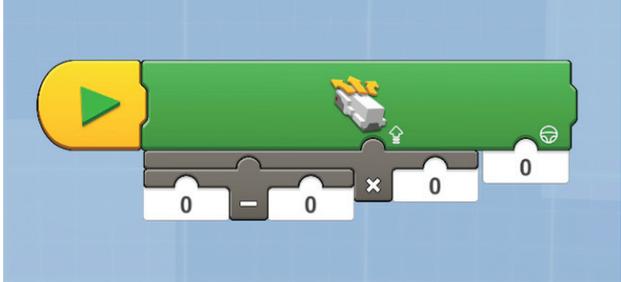
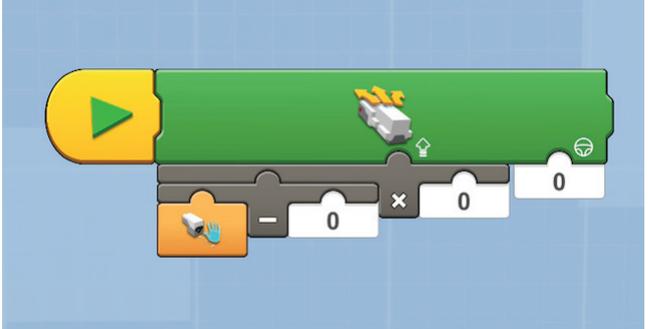
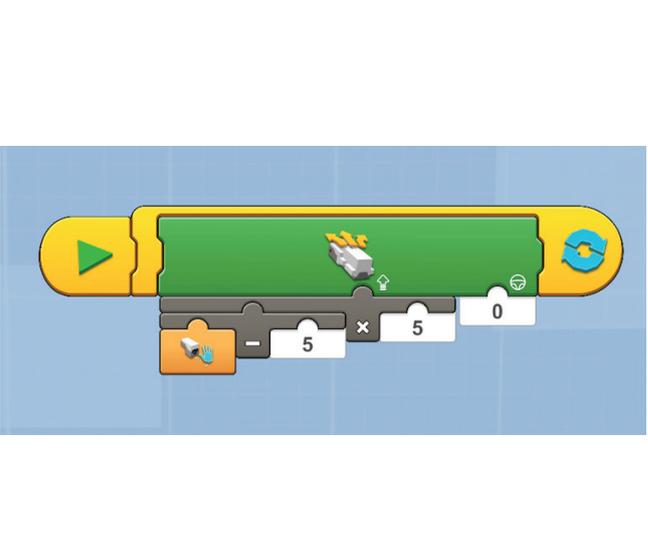
For this task, error will be how far the robot is from the target distance and correction will be moving the robot forward or backward at a particular speed.

### Robot Design:

Build any design you like with two wheels or treads such that the robot can move forward and backwards. Be sure to include the Color & Distance Sensor somewhere on the front of your design.



## Programming Steps:

<p><b>Step 1:</b> Drag in a 'Start' Block from the Yellow Palette</p> <p><b>Step 2:</b> Drag in a 'Motor On' Block from the Green Palette and place near Start Block.</p>	
<p><b>Step 3:</b> Drag in the 'Multiplication' Block from the White Palette and place as indicated. This will be the part of the code that computes the proportional response to the distance from the target (i.e. the correction).</p>	
<p><b>Step 4:</b> Drag in the 'Subtraction' Block from the White Palette and place as indicated on the left parameter of the Multiplication Block. This part of the code will compute the distance from the target (i.e. the error)</p>	
<p><b>Step 5:</b> Drag in the 'Distance Sensor' from the Cyan Palette and place as indicated in the left parameter of the Subtraction Block.</p>	
<p><b>Step 6:</b> Change all the parameters to 5. The 5 in the Subtraction Block sets the target because the range of the sensor is 0-10 and 5 is halfway in between. The Multiplication Block scales the power so that instead of going 0–5 power, the robot will go 0-25 power. When you take the output from the sensor and subtract 5, the result may be a negative number. Since negative power makes the motor move backwards, the robot will move away from the obstacle.</p> <p><b>Step 7:</b> Drag in a Loop Block from the Yellow Palette so that the code is repeated again and again forever.</p>	

#