ADVANCED EV3
PROGRAMMING LESSON

Squaring or Aligning on a Line

By Droids Robotics

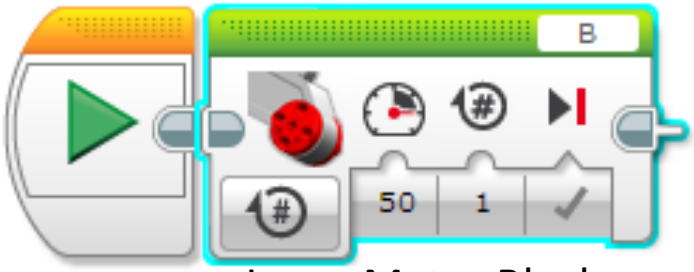
Lesson Objectives

- Learn how to get your robot to square up (straighten out) when it comes to a line
- Learn how squaring (also known as aligning on a line) can help the robot navigate
- Learn how to improve initial code for aligning by repeating a technique
- Practice creating a useful My Block

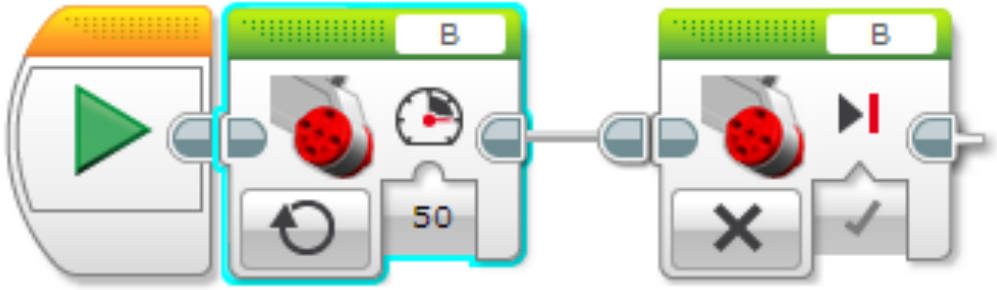
- Prerequisites: My Blocks with Inputs & Outputs, Data Wires, Parallel Beams, Parallel Beams Synchronization

Review: Motor Movements

- Move Steering lets you control both motors at the same time
- What if you want to move or stop one motor at a time?
  - Use the Large Motor Block



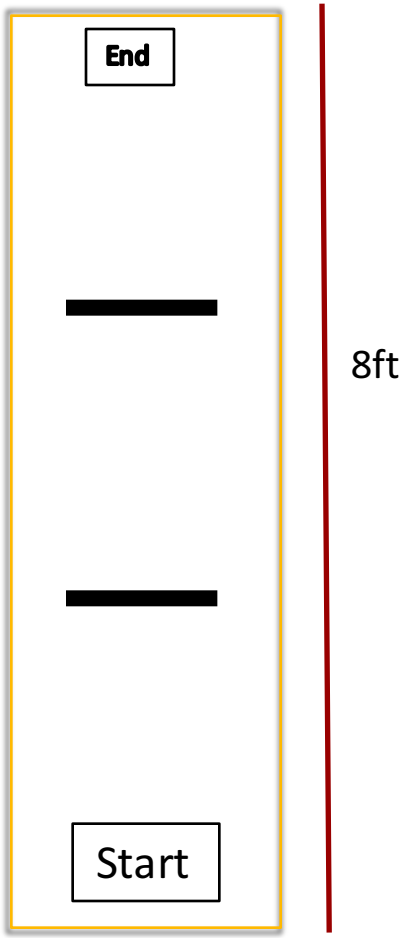
Large Motor Block



Large motor block in ON mode / OFF mode

Why Align on a Line?

- Aligning on a line helps the robot navigate
  - Robots get angled as they travel farther or turn (the error accumulates)
  - Aligning on a line can straighten out a robot.
  - Aligning can tell a robot where it is when it has to travel far
- Example Goal: Your robot must deliver an object only inside a small END area. The distance between start and end is 8 feet
  - Do you think your robot can travel 8 feet and continue to be straight?



Three Easy Steps to Align

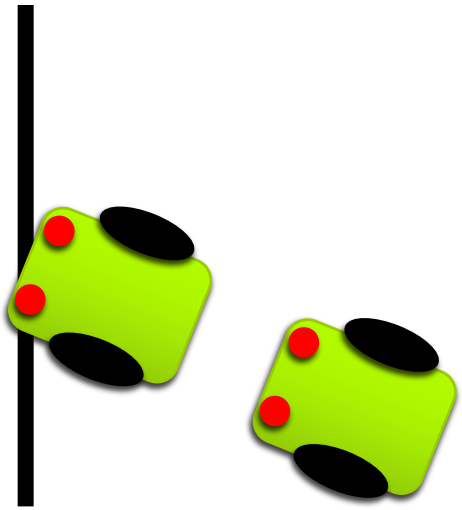
Challenge: Make the robot straighten out (align/square up)

STEP 1: Start both motors

STEP 2: Stop one motor when the sensor on the corresponding side sees the line

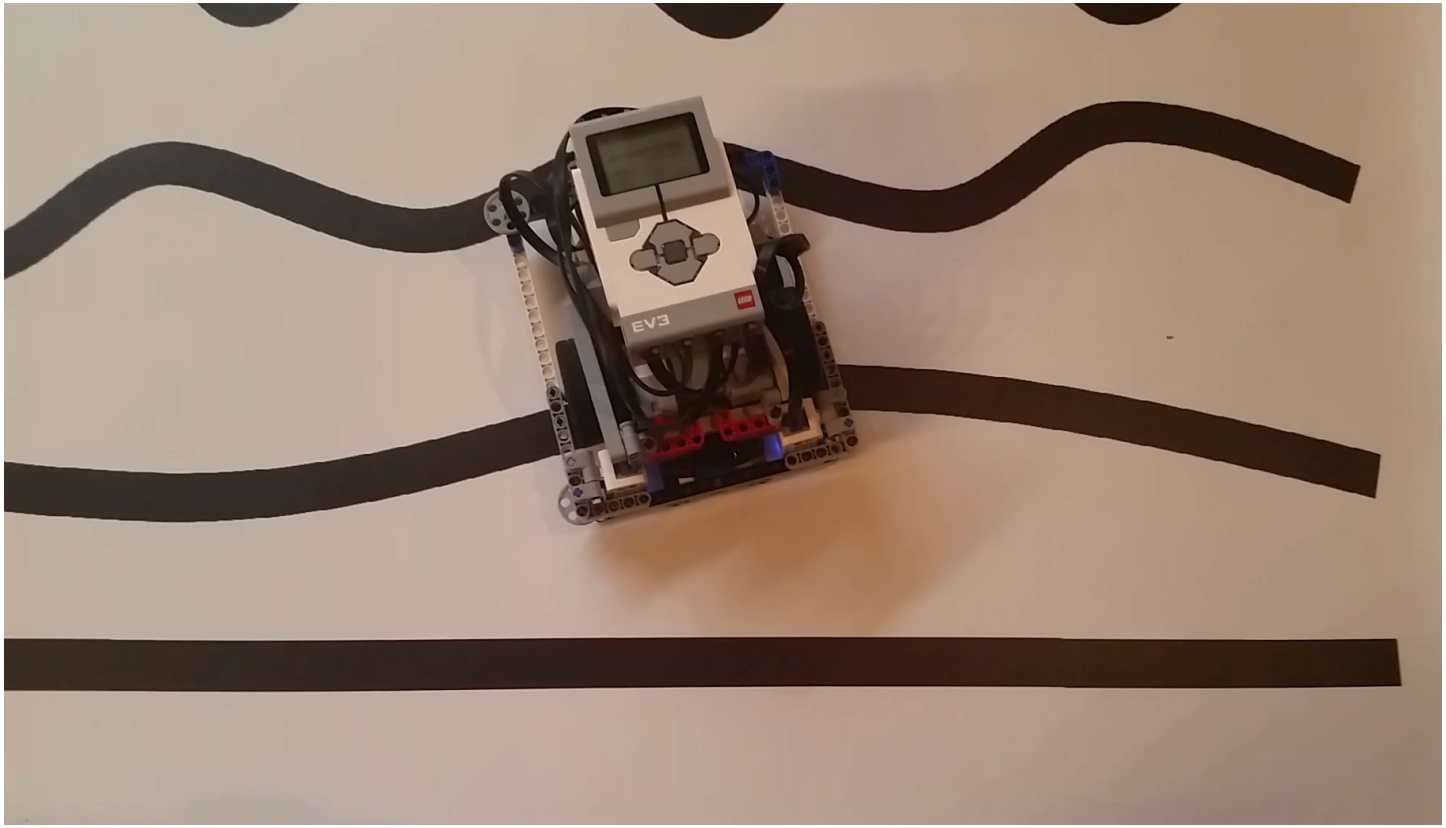
STEP 3: Stop moving the second motor when the sensor on that side sees the line

Hints: Use a Large Motor Block, Use Parallel Beams, Use the Large Motor Block



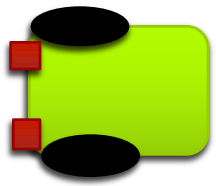
(This slide is animated)

What Aligning Should Look Like



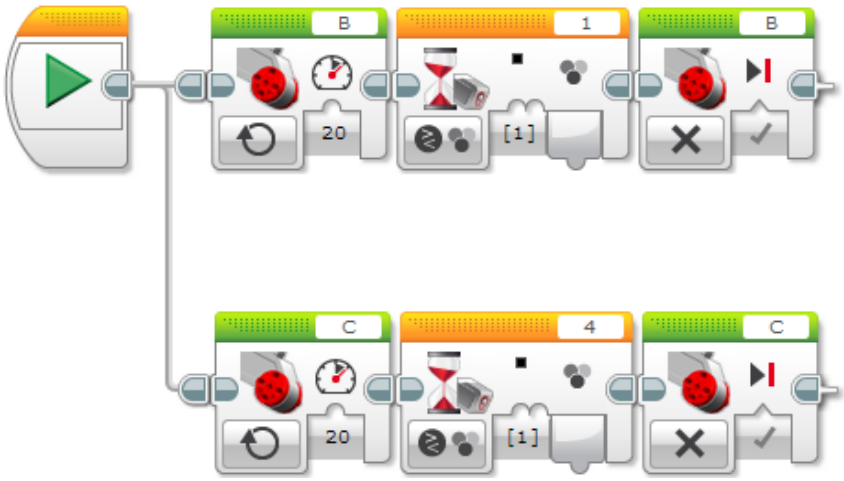
Notes About Our Solution:

- Our solution uses 2 Color Sensors (connected in Ports 1 and 4).
- Our solution assumes that the color sensor on port 1 is next to the wheel on motor port B and color sensor on port 4 is next to the wheel on motor port C.
- You should adjust the ports as needed
- Your color sensors should NOT be placed right next to each other (See red boxes below in robot image. These are the color sensors.)



Basic Solution: Moving Until Line

We use a parallel beam here so that we can do 2 simultaneous actions.
In the top beam, Motor B on our robot keeps moving until the Color Sensor on Port 1 sees Black. Then it stops.
In the bottom beam, Motor C on our robot keeps moving until the Color Sensor on Port 4 sees Black. Then it stops.



Step 1 Goal: Create a SIMPLE way to square up on the line

Note 1: You will need 2 EV3 Color Sensors (connected in Ports 1 and 4 in this program)

Note 2: This program squares onto a Black Line (you can change this to whatever color the EV3 accepts.

Note 3: This program uses the color sensor in COLOR MODE. You can write a program that uses LIGHT MODE, but you will have to calibrate your sensors. We will show you that in another lesson.

Note 4: Your robot design will make a difference - whether you have your color sensors in the rear or front of your robot, and how far apart the sensors are (the further apart, the better).

Note 5: You should adjust the ports as needed - e.g. this assumes that a color sensor on port 1 is next to the wheel on motor port B and color sensor on port 4 is next to the wheel on motor port C.

Note 6: While the robot will be on the black line, this will not create a perfect alignment. See instruction in Step 3 for a simple fix.

Note: Synchronization & Parallel Beams

- When you have two or more beams you do not know when each beam will finish.
- If you wanted to move after the align finishes you might try to add a move block at the end of one of the beams.
  - Note: This will not work because EV3 code will play your move block without waiting for the other beam to finish.
  - Solution: You need to synchronize your beams. To learn more about synchronization and solutions go to the Advanced EV3Lessons.com Lesson on Sync Beams
- The problem of synchronization can also be solved by making a My Block out of the align code (refer to My Block lesson in Intermediate)
  - My Blocks always wait for both beams to finish before exiting

Improving Your Align Code

- What do you notice about the solution we just presented?
  - The robot isn't quite straight (aligned) at the end of it.
  - Both color sensors are on the line, but the robot stops at an angle.
- Challenge Continued: Think about how you can improve this code so that the robot ends straighter
  - Hint: Can you repeat the last process by looking for **white**?
  - This assumes that the black line we were straightening out on has white on both sides.

Tips for Success

- You will get better results
  - ….if your color sensors are about 1cm from the ground
  - ….if you shield your color sensors
  - ….if you don’t come at the line at steep angles
  - ….if you keep your color sensors spread apart

Discussion Guide

1. What does Aligning on a Line/Squaring mean?
   Ans. It means when your robot comes at an angle, it can use a line to straighten out

2. Why does Squaring help when you travel long distances?
   Ans. Robots do not move accurately/straight. Squaring can help you know where you are and straighten out.

3. Why might it be useful to repeat the alignment process multiple times?
   Ans. Your robot will get straighter each time you repeat

Credits

- This tutorial was created by Sanjay Seshan and Arvind Seshan from Droids Robotics.
- More lessons at www.ev3lessons.com