PROGRAMMING YOUR EV3 IN 10 EASY LESSONS

By: Droids Robotics
www.ev3lessons.com

SECTION 1: EV3 BASICS

THE “BRICK” BUTTONS

1 = Back

Undo

Stop Program

Shut Down EV3

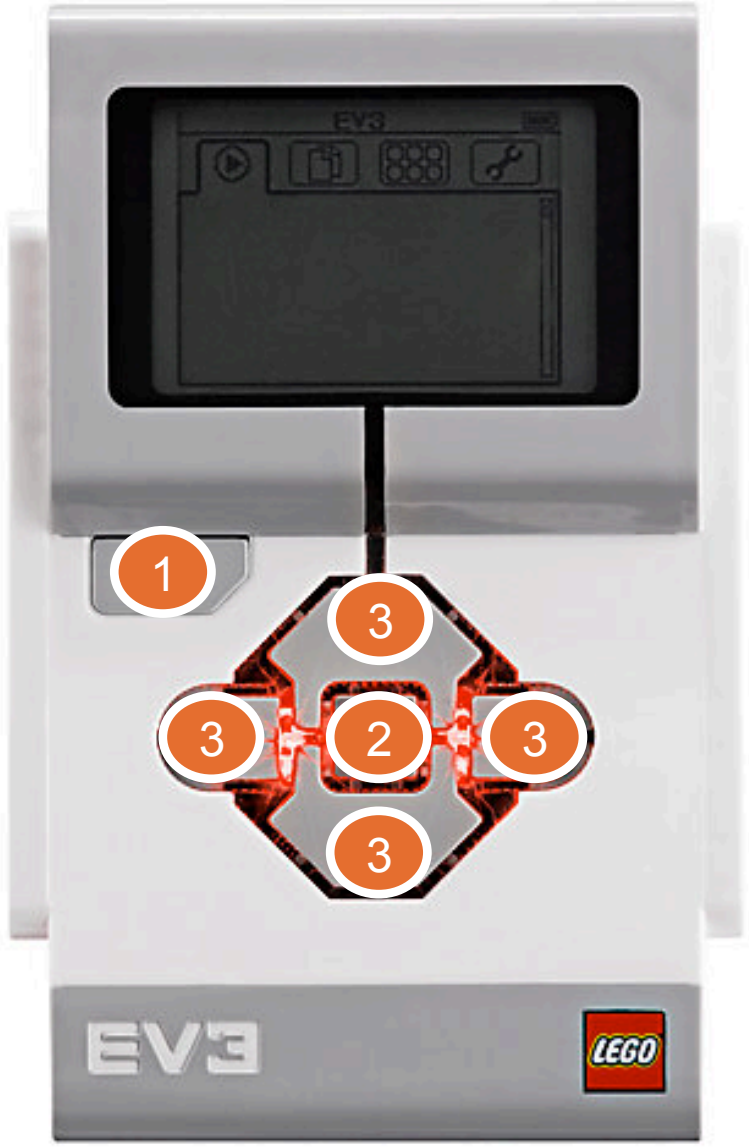
2 = Center Button

Select options

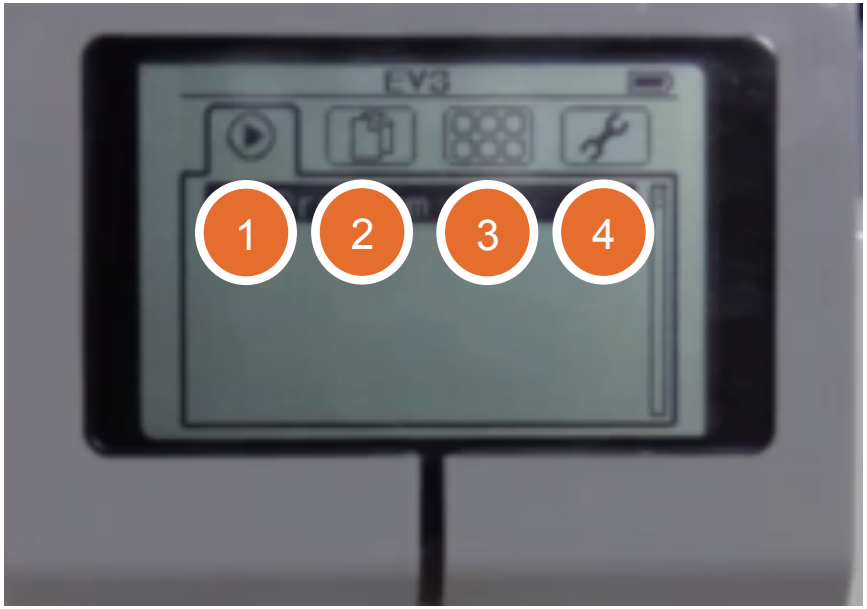
Run Program

3 = L, R, Up, Down

Navigate menus



THE “BRICK” SCREEN



Tabs on Screen

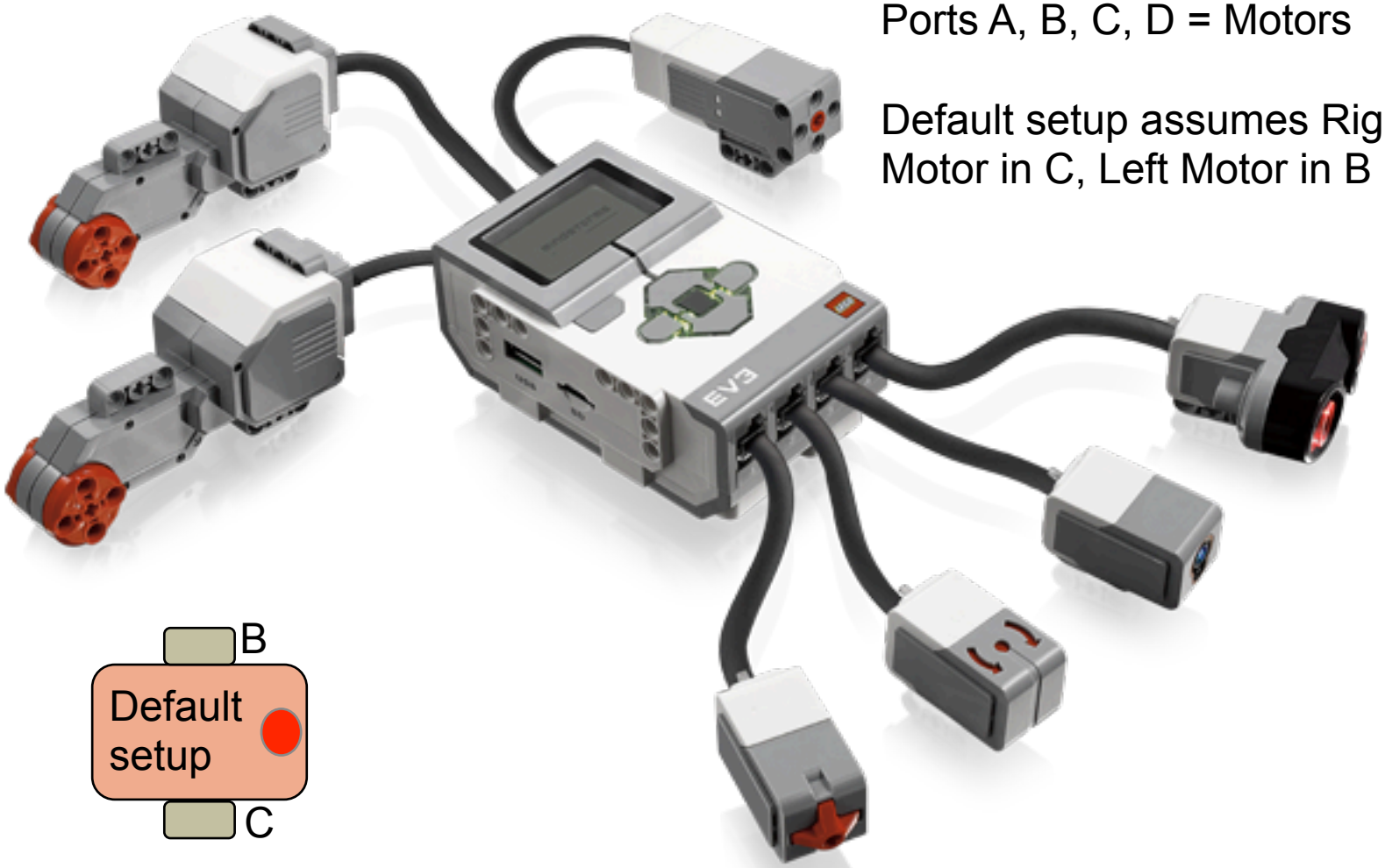
1. **Run Recent**
   Find programs you ran recently

2. **File Navigation**
   Find all programs by project

3. **Brick Apps**
   Port views

4. **Settings**
   Bluetooth, Wifi, Volume

PORTS, SENSORS, MOTORS

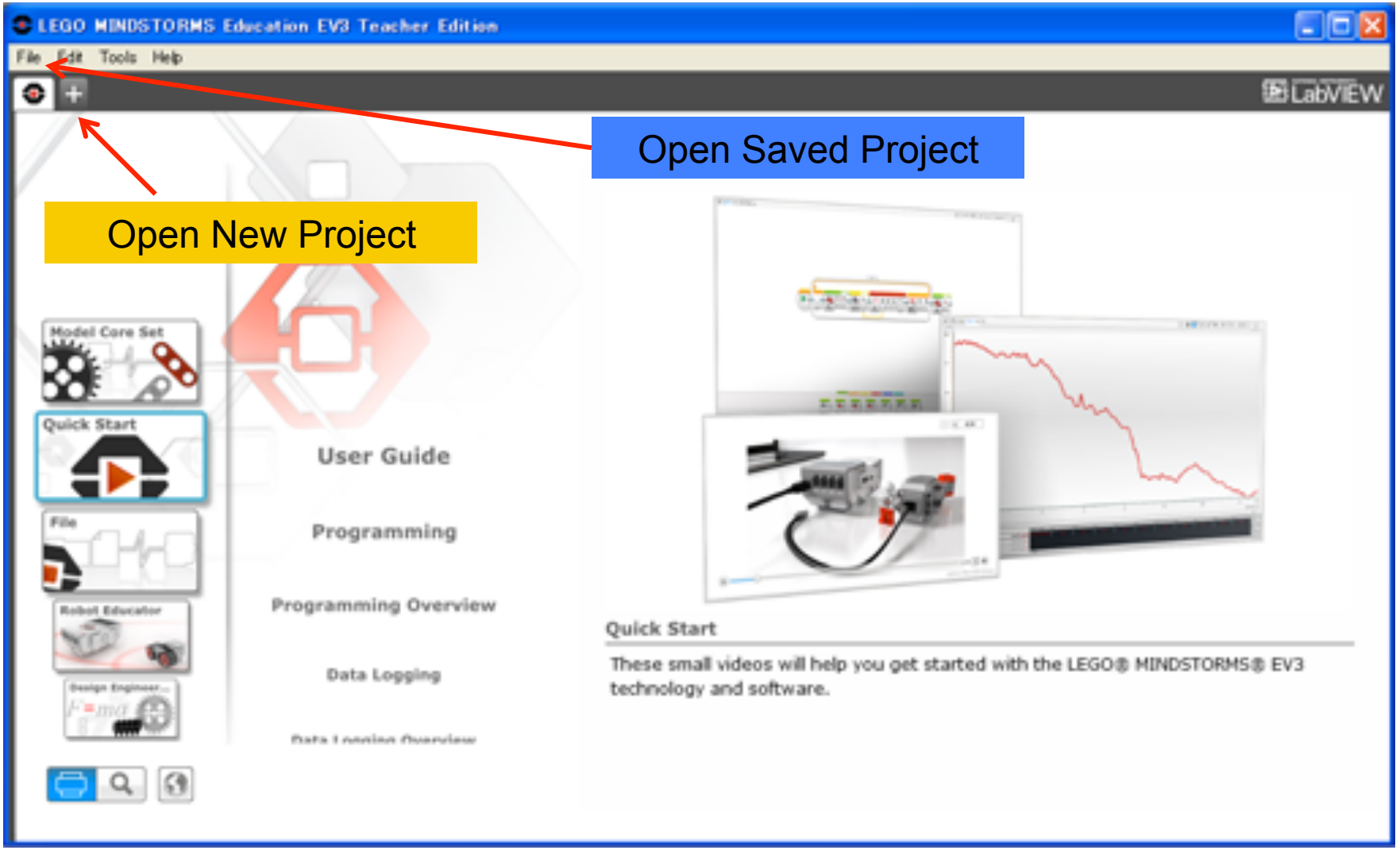


Ports A, B, C, D = Motors

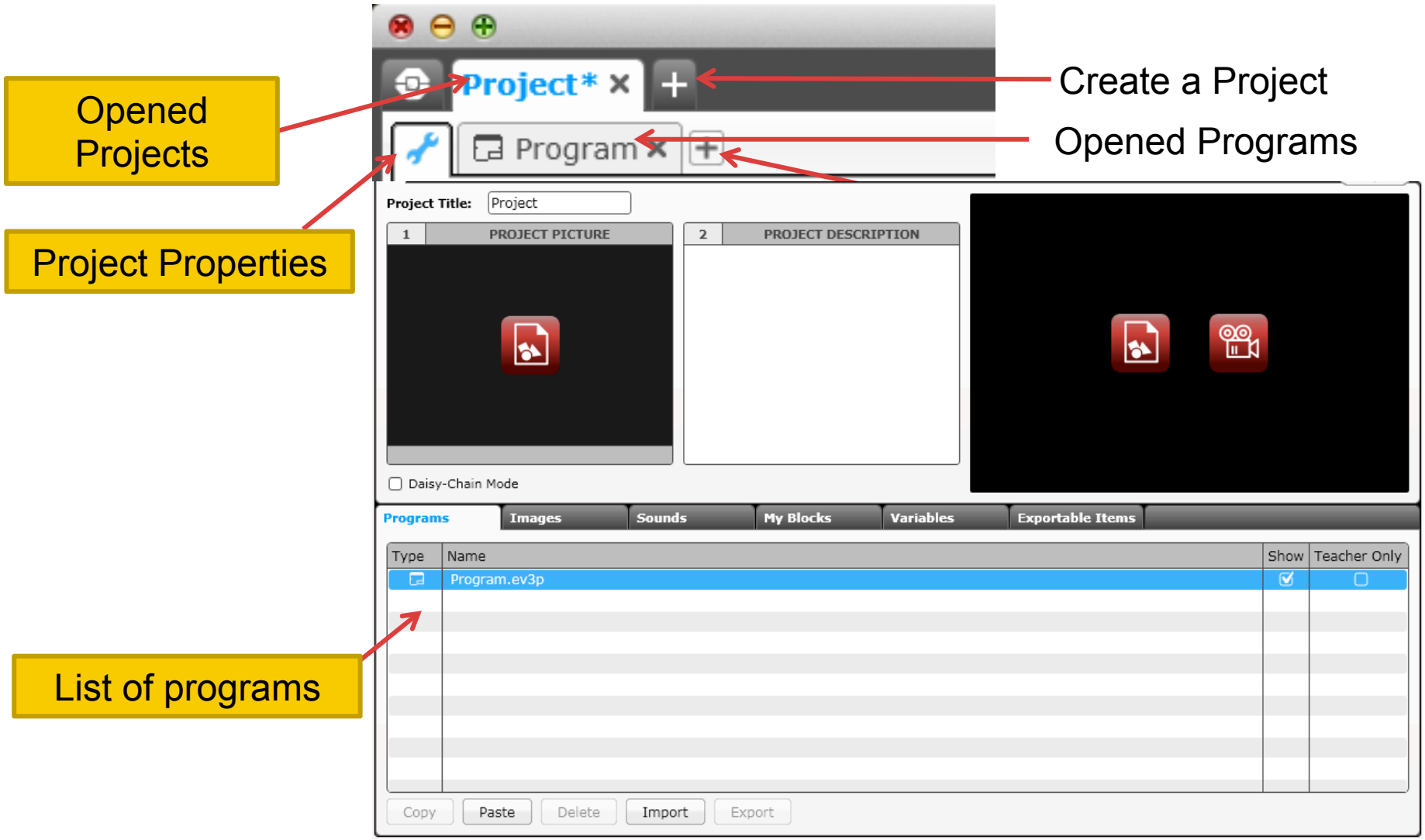
Default setup assumes Right Motor in C, Left Motor in B

Ports 1, 2, 3, 4 = Sensors

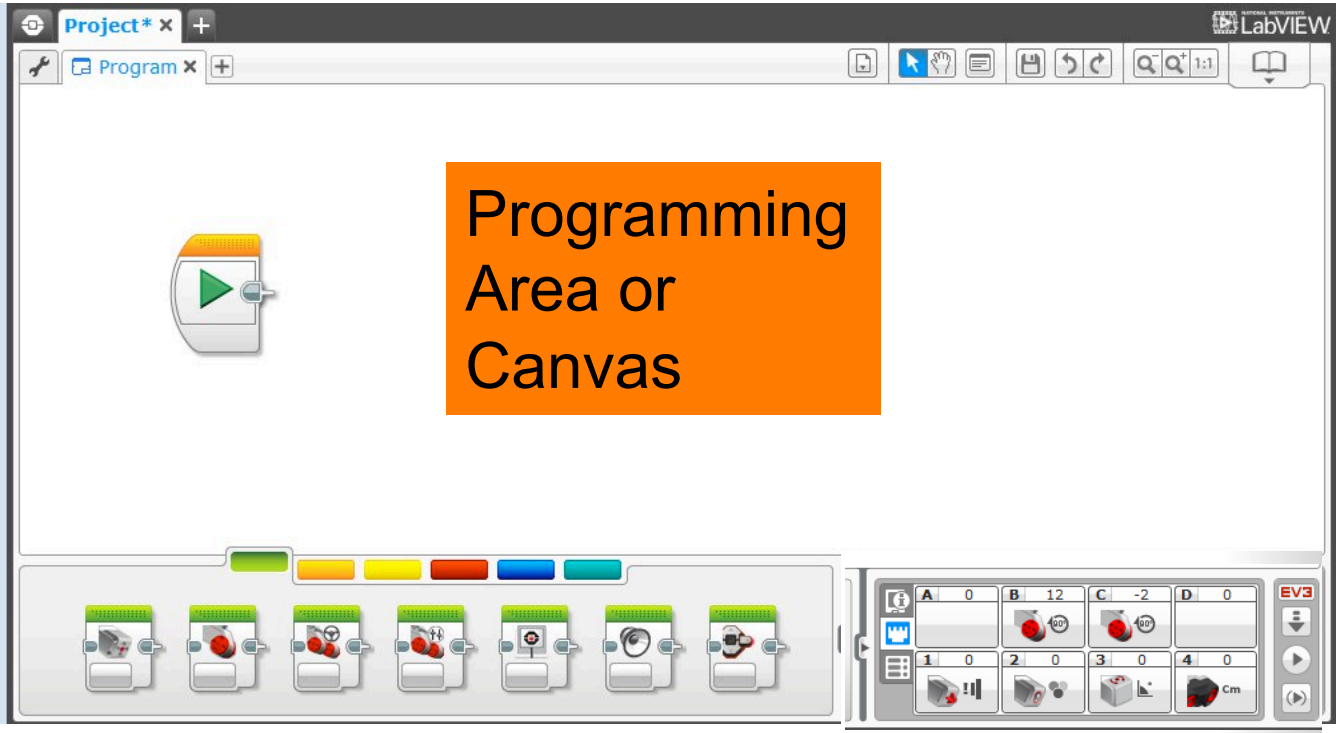
EV3 SOFTWARE



EV3 SOFTWARE: STARTING A NEW PROGRAM



EV3 SOFTWARE: PROGRAMMING SCREEN



Programming Blocks in 6 Colored Tabs

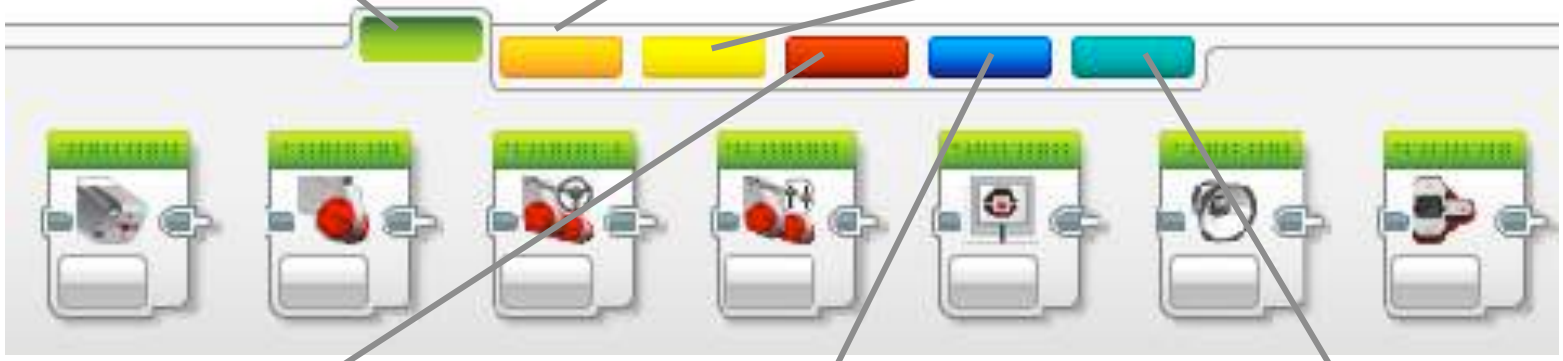
Brick Status & Downloading

EV3 BLOCKS: COLORED TABS

ACTION BLOCKS
Move, Large & Medium Motor, Display…

FLOW BLOCKS
Start, Wait, Loop, Switch, Loop Interrupt

SENSOR BLOCKS
Brick Buttons, Gyro, Color, Ultrasonic



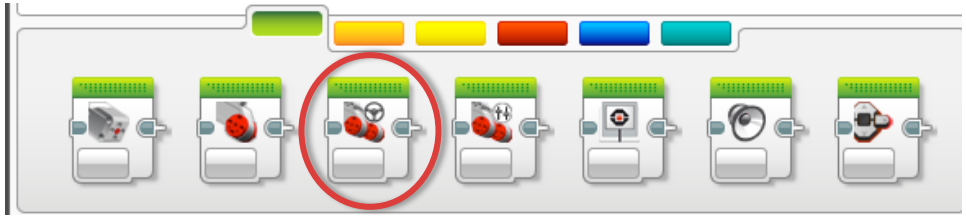
DATA OPERATIONS
Variables, Array, Logic, Math, Compare…

ADVANCED BLOCKS
Data Logging, Unregulated Motor…

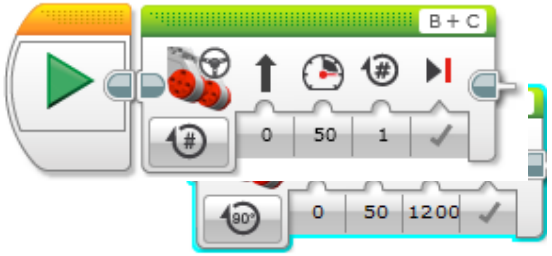
MY BLOCKS
Custom Blocks you create

SECTION 2: MOVING STRAIGHT

CHALLENGE: MOVE STRAIGHT

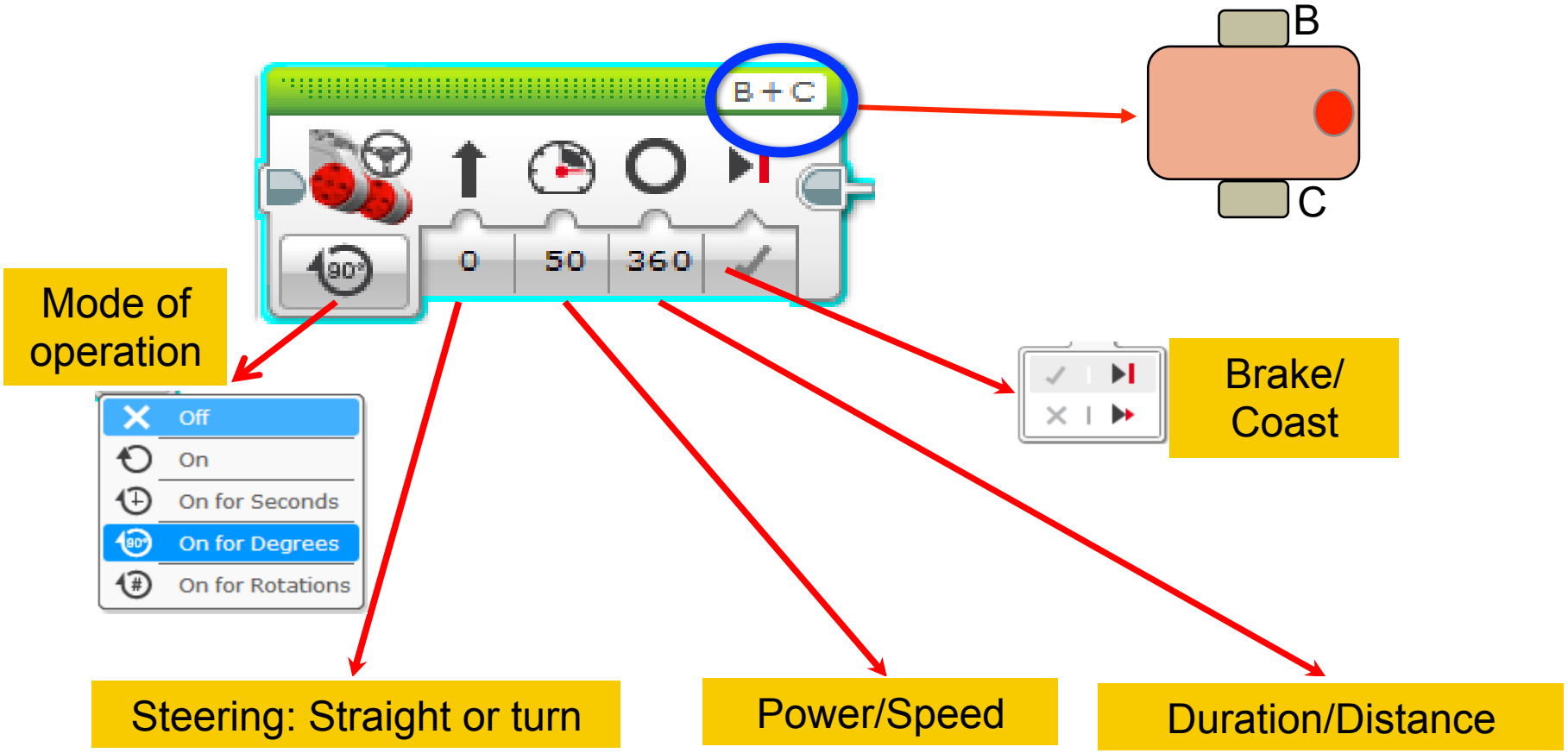


STEP 1: Green Block Tab, Click and hold Move Steering and drag to programming area

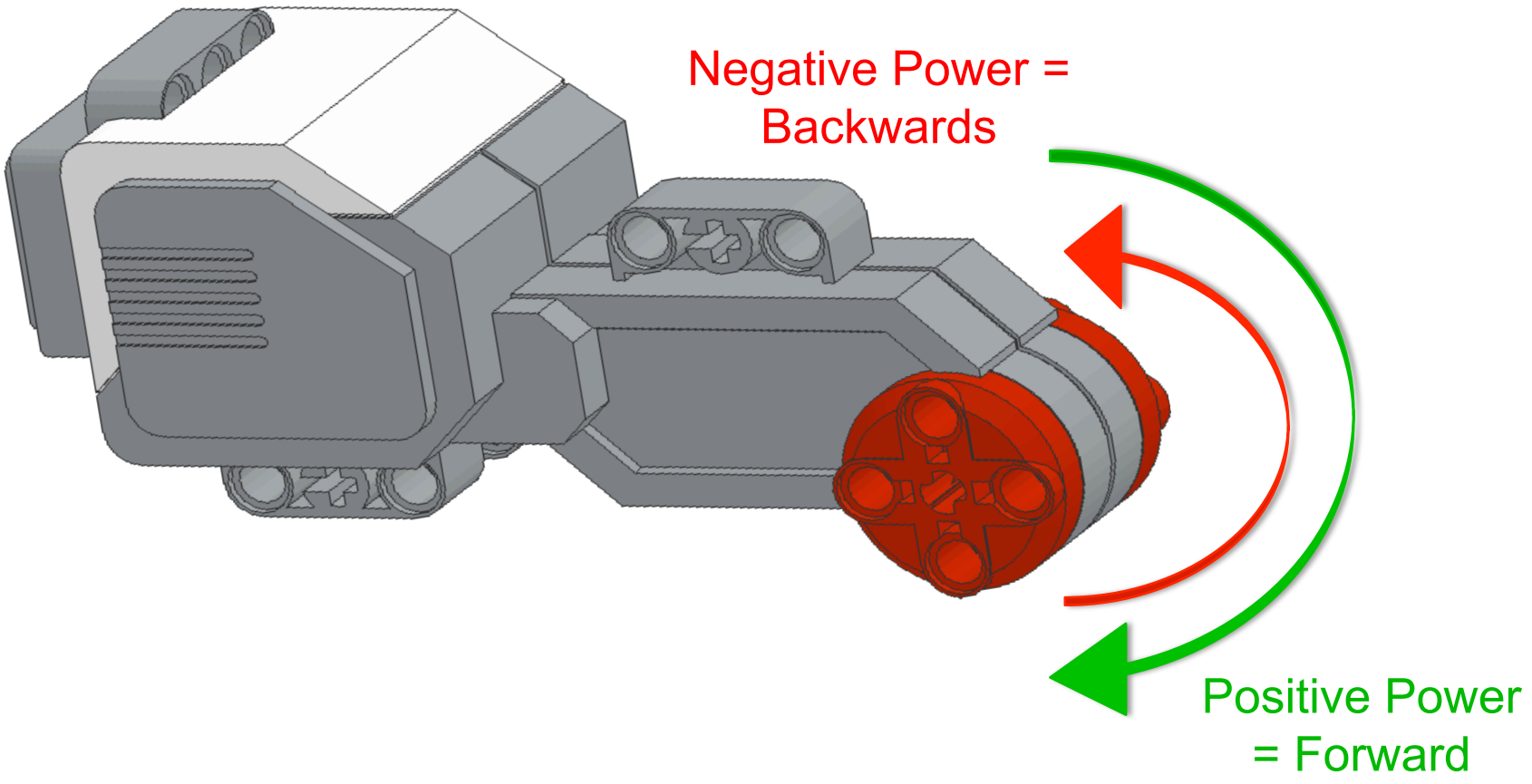


STEP 2: Drop next to the Start Block (green arrow)

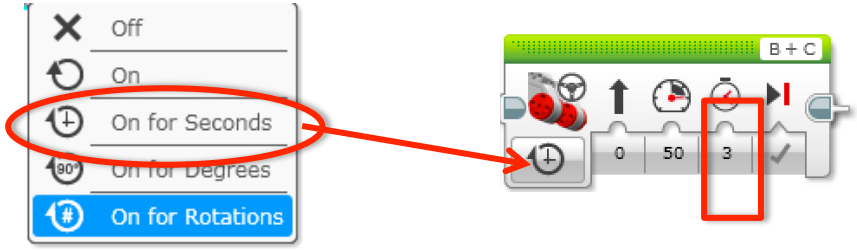
MOVE STEERING BLOCK

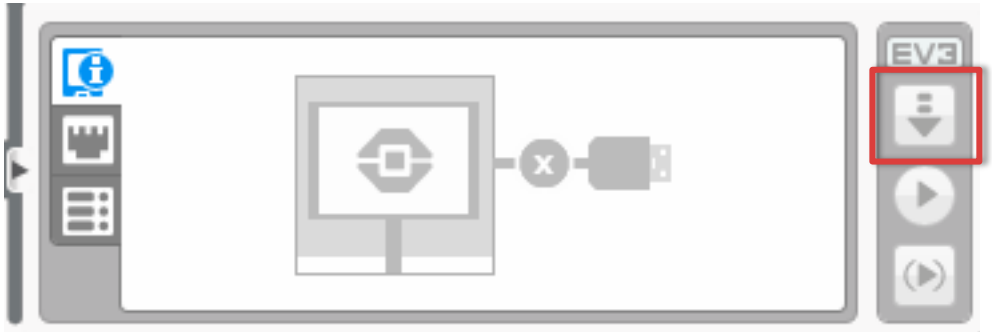


NEGATIVE & POSITIVE POWER: BACKWARD & FORWARD



CHALLENGE: MOVE STRAIGHT (3 SECONDS)





STEP 1: Green Block Tab, Click and hold Move Steering and drag to programming area

STEP 2: Drop next to the Start Block (green arrow)

STEP 3: Select Options. Move “3 Seconds”

STEP 4: Connect USB cable to EV3 and Laptop.

STEP 5: Download to EV3

MOVE STRAIGHT: SECONDS VS. DEGREES VS. ROTATIONS

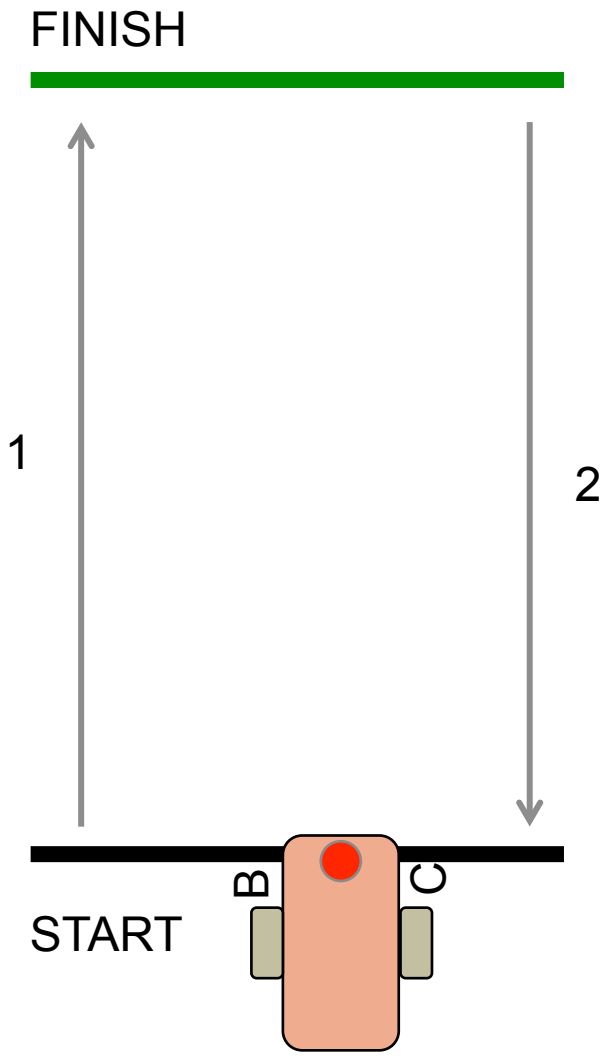
CHALLENGE: Move your robot forward from the start line to the finish line (1) and back to the start (2).

Try: Select SECONDS, DEGREES or ROTATIONS

Try: Different speeds

What are some advantages or disadvantages of using each one?

Did you have to do a lot of guess and check to reach the line?



MOVE STRAIGHT CHALLENGE – PART 1 SUMMARY

TEAMS	SECONDS	DEGREES	ROTATIONS
Team 1			
Team 2			
Team 3			
Team 4			

MOVE STRAIGHT DISCUSSION

Teams	Did you guess and check a lot?	Did changing the speed matter?	What was more accurate? Seconds? Degrees?	Do you think wheel size will matter?	Do you think battery level matters?
Team 1					
Team 2					
Team 3					
Team 4					

MOVING STRAIGHT MORE ACCURATELY: USE PORT VIEW

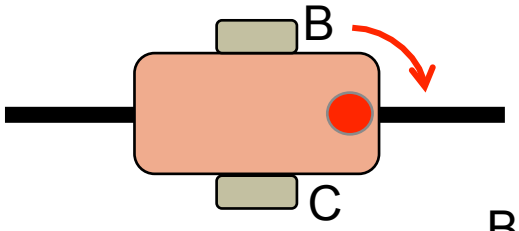
Try "port view" on brick (on Brick Apps tab)

- Move your robot with your hand from your start line to your end line
- Read how many degrees your robot moved
- Use this number in the Move Steering Block to move the correct distance.

SECTION 3: TURNING

PIVOT VS. SPIN TURNS

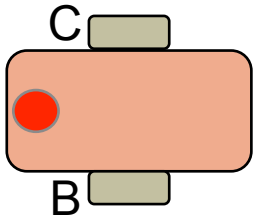
180 Degree Pivot Turn



Start Position

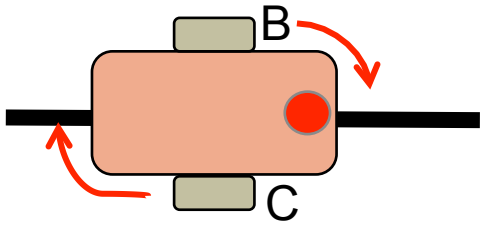
Motor
B Moves

End Position



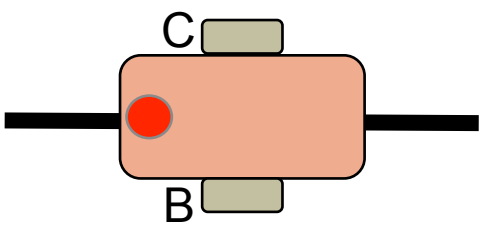
180 Degree Spin Turn

Start Position



Motors
B and C
Move

End Position

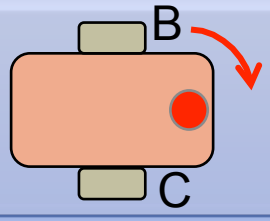
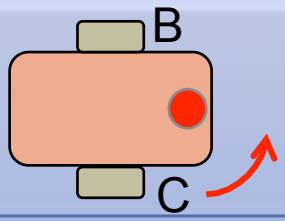
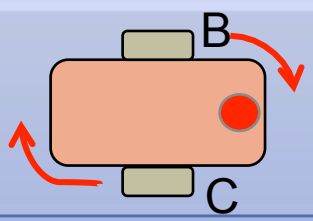
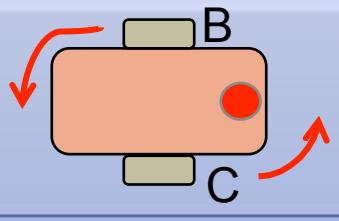


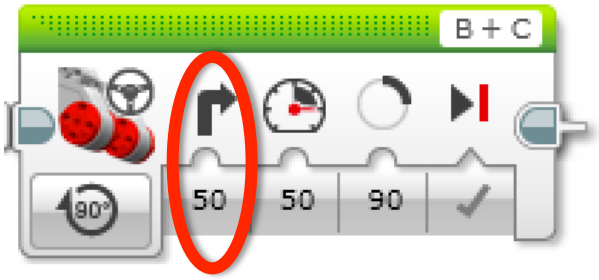
Notice where the robot ends in both pictures after a 180 degree turn.

In the Spin Turn, the robot moves a lot less and that makes Spin Turns are great for tight positions. Spin turns tend to be a bit faster but also a little less accurate.

So when you need to make turns on the FLL board, you should decide which turn is best for you!

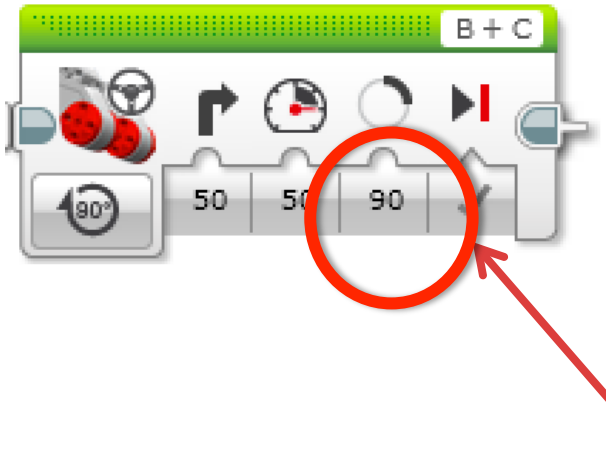
HOW TO MAKE PIVOT AND SPIN TURNS

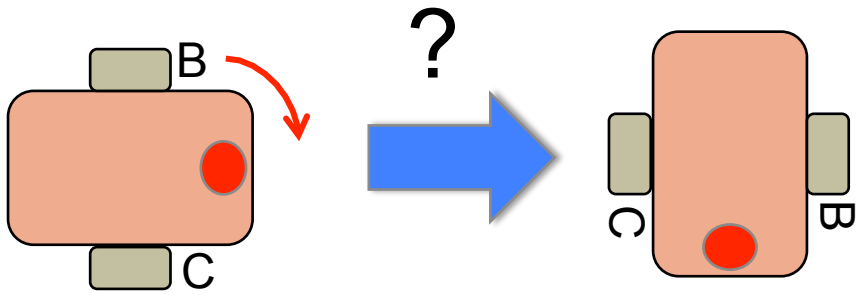
Steering Value			
50	-50	100	-100
			
Pivot Turn Right	Pivot Turn Left	Spin Turn Right	Spin Turn Left



Change Steering value here

MAKING A PIVOT TURN FOR 90 DEGREES

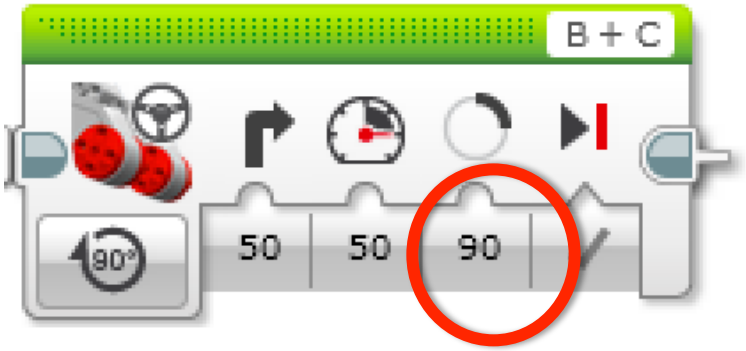




Program your robot to turn 90 degrees....Does the robot actually turn 90 degrees if you just pick 90 degrees for distance?

HOW DO YOU MAKE THE ROBOT TURN 90 DEGREES?

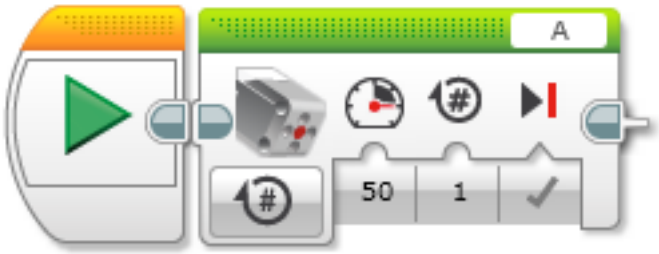
Ans. Try using the port view again to measure the turn and then input the correct number of degrees.



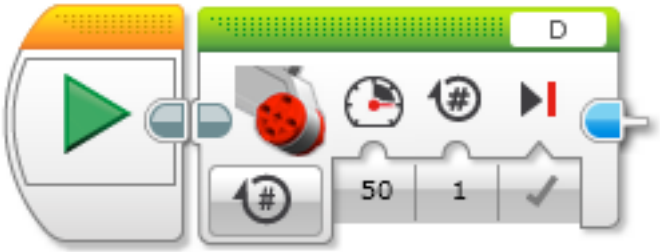
TURNING AN ATTACHMENT ARM, NOT JUST THE WHEELS

- **Attach a medium motor to Port A or a large motor to Port D as needed.**
- **Move Steering vs. Motor Block**
  - For moving your wheels you should use a Move Steering Block that syncs both wheel motors.
  - For moving your attachment your arm, you use either a Medium Motor Block or a Large Motor Block because you don't need to sync your motors.

Medium Motor Block



Large Motor Block

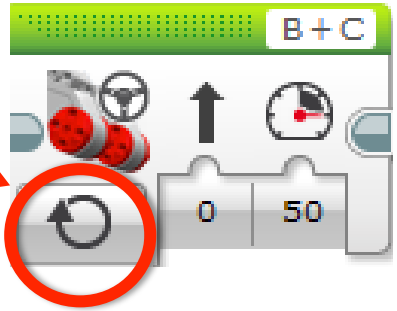


MOTOR "ON" AND "WAIT" BLOCKS

Leaving the motor "on" and "off"

Why use the "on" instead of "degrees"?

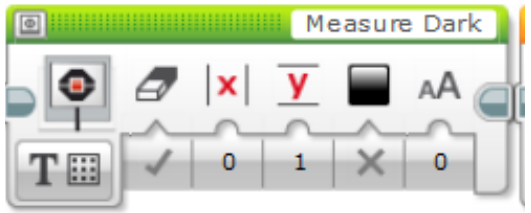
- May want the program to do other tasks while moving



"Wait" block in Flow Tab

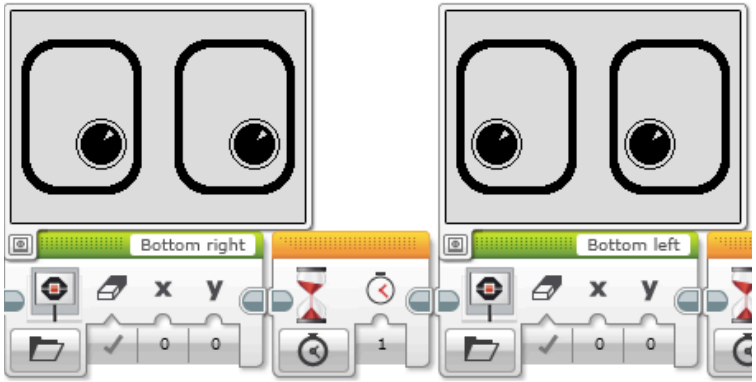
- Wait for seconds

SOMETHING FUN: DISPLAY BLOCK

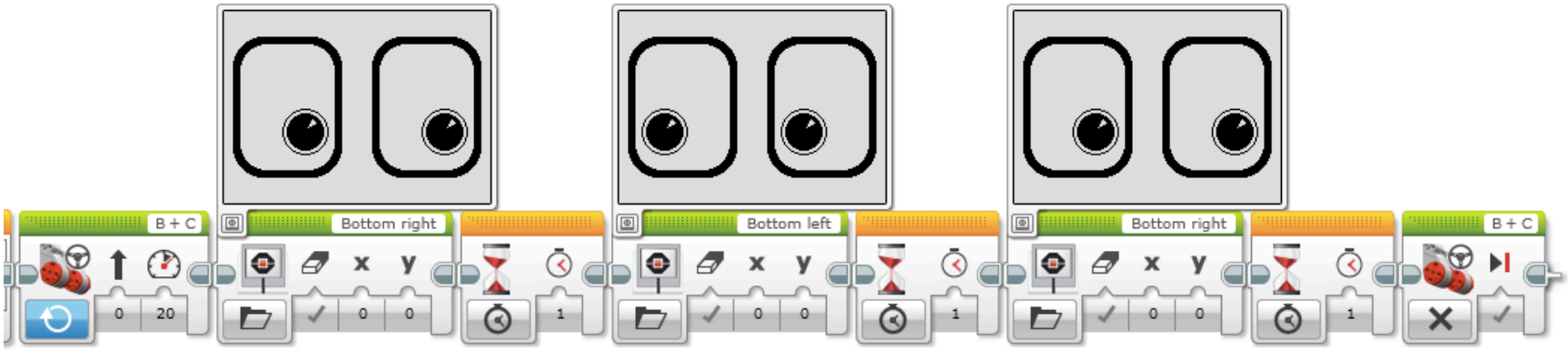


- Use the Display Block to display information and pictures on the screen
- You can control the location and size of text
- You can use this same block to display sensor readings and instructions.

- Challenge: Can you display eyes on the screen while moving? Alternate eyeballs that look left and right.
  - Use the Display Block, Motor On and Wait Block



CHALLENGE SOLUTION



Motor On

Display Block

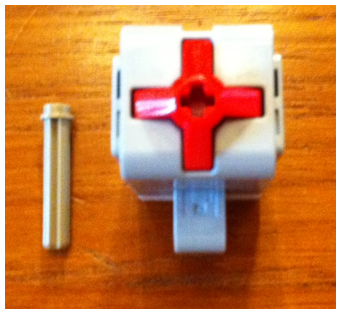
Wait Block

Motor Off

SECTION 4: TOUCH SENSOR

MODIFICATION TO YOUR ROBOT BUILD

If you built the robot that we recommended, you will have to make a minor modification to the design so that the touch sensor is more accessible.



SENSOR TAB VS. FLOW TAB (READING VS WAIT FOR SENSOR)

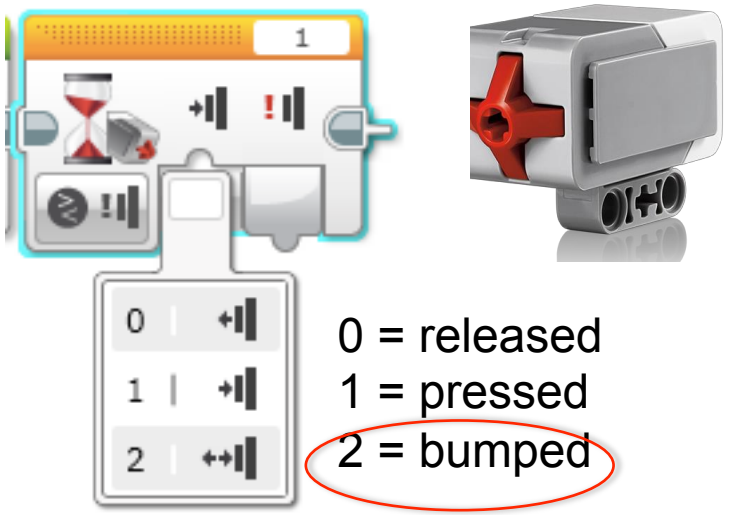
Compare sensor block to wait block

- When should we use one or the other
- Sensor Tab Sensor Blocks = Reading and Comparing Sensor Values
- Flow Tab Wait For Block = Wait until a sensor reading

In this tutorial, we will use the Wait For Block

USING THE TOUCH SENSOR

Challenge 1: **Program your robot to move straight until you touch the sensor with your hand.**



Challenge 2: **Program your robot to move until it hits the edge of the FLL table past the dog and cat missions. Then back up to the black line and turn right 90 degrees.**

Hint: You will combine everything you have learnt so far: Move Steering + Turning + Wait Block

SECTION 5: COLOR SENSOR

USING THE COLOR SENSOR

One of the ways for the robot to know its location is to take advantage of the markings on the field mat.

Every year, the Robot Game's mat has lines or dark markings that can be detected by the Color Sensor.

Common uses:

- Move until a line
- Follow a line

Introduce color sensor

Modes:

- Color, Reflected Light, Ambient Light
- We will use the COLOR mode in this tutorial.

USING COLOR SENSOR IN COLOR MODE

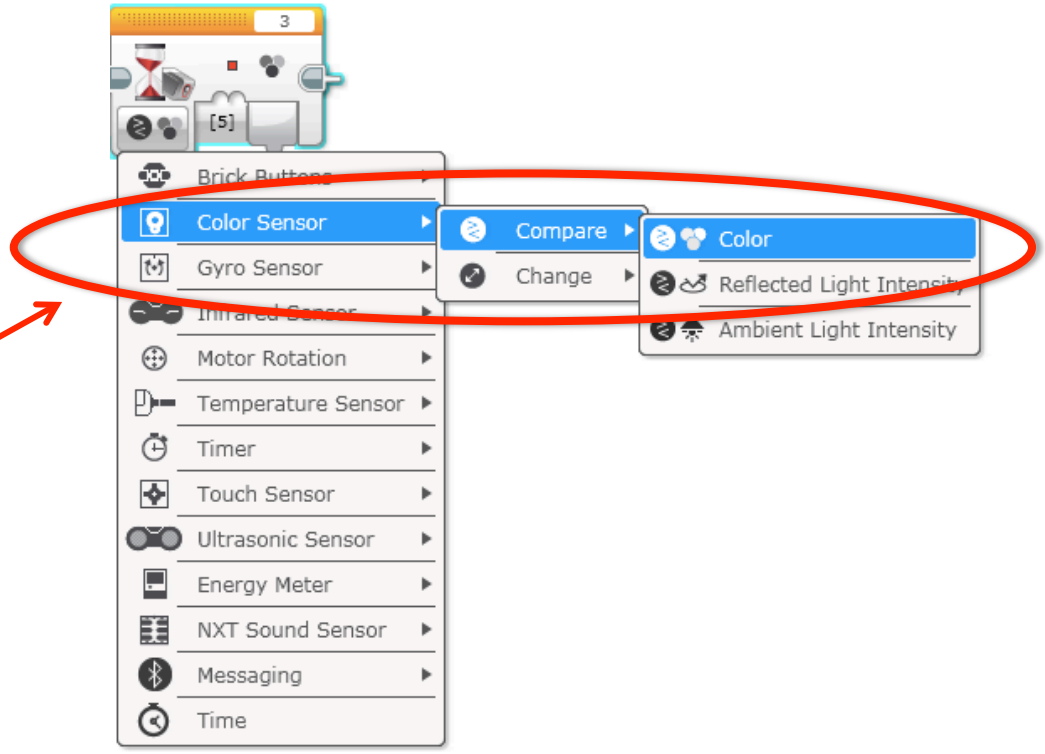
CHALLENGE: Make the robot move up to the green line using the color sensor?

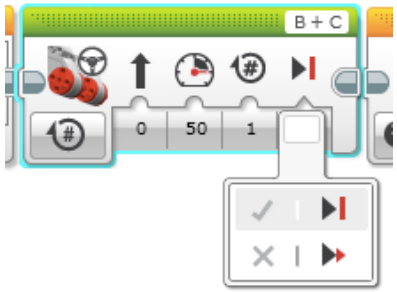
Step 1: Use Wait For Color

Step 2: Use the color sensor in COLOR MODE

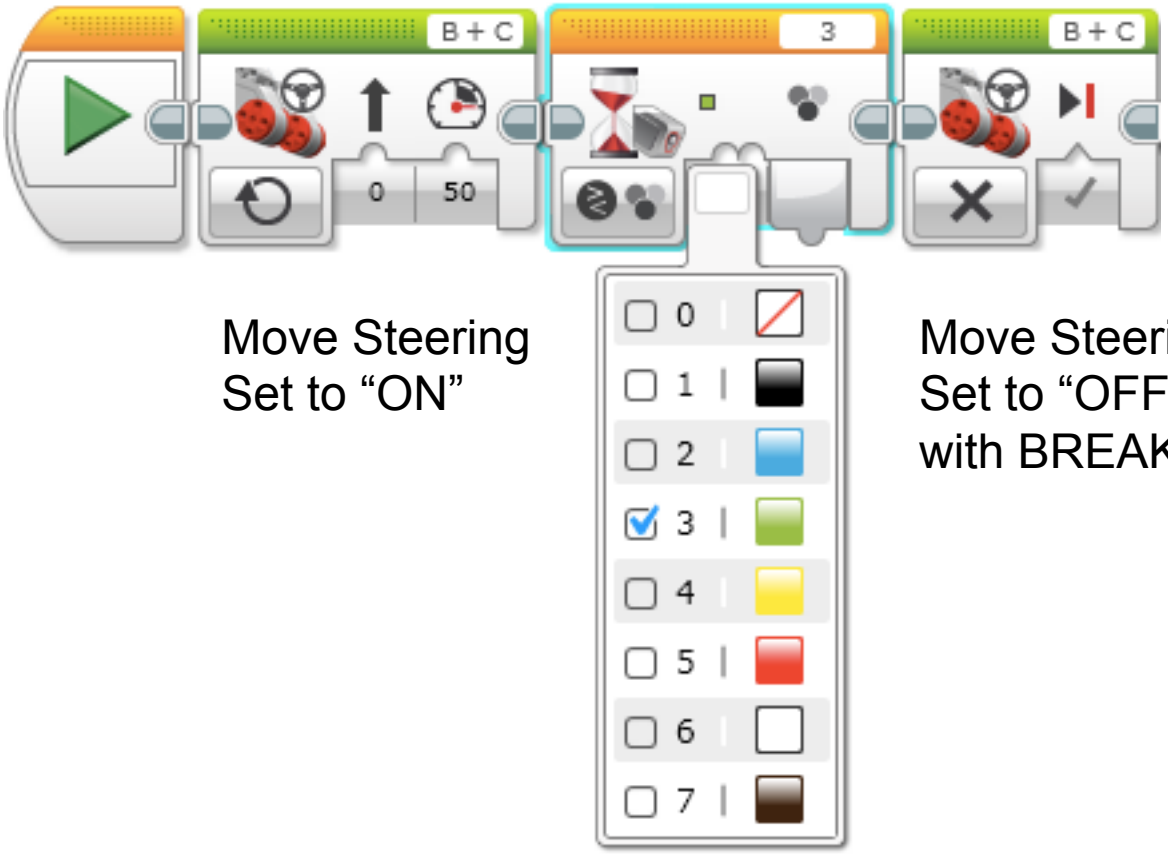
Step 3: Coast or Break?

Tip: Coast will make the motors keep moving. Break makes the motors stop immediately. Which do you use to stop EXACTLY on the green line?





COLOR SENSOR CHALLENGE SOLUTION



Move Steering
Set to “ON”

Move Steering
Set to “OFF”
with BREAK

Wait until Color is Green (#3)

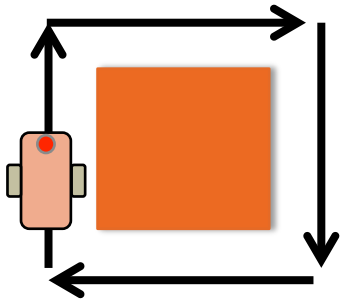
FINISH

START

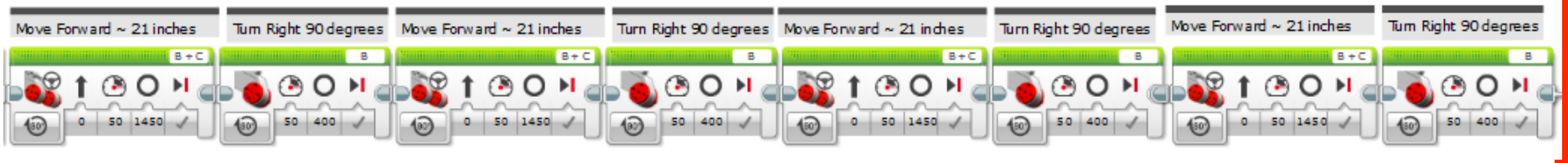
SECTION 6:
REPEATING ACTIONS

REPEATING AN ACTION

How can we move around a box using the commands we already know?



- **(move + turn) + (move + turn) + (move + turn) …**



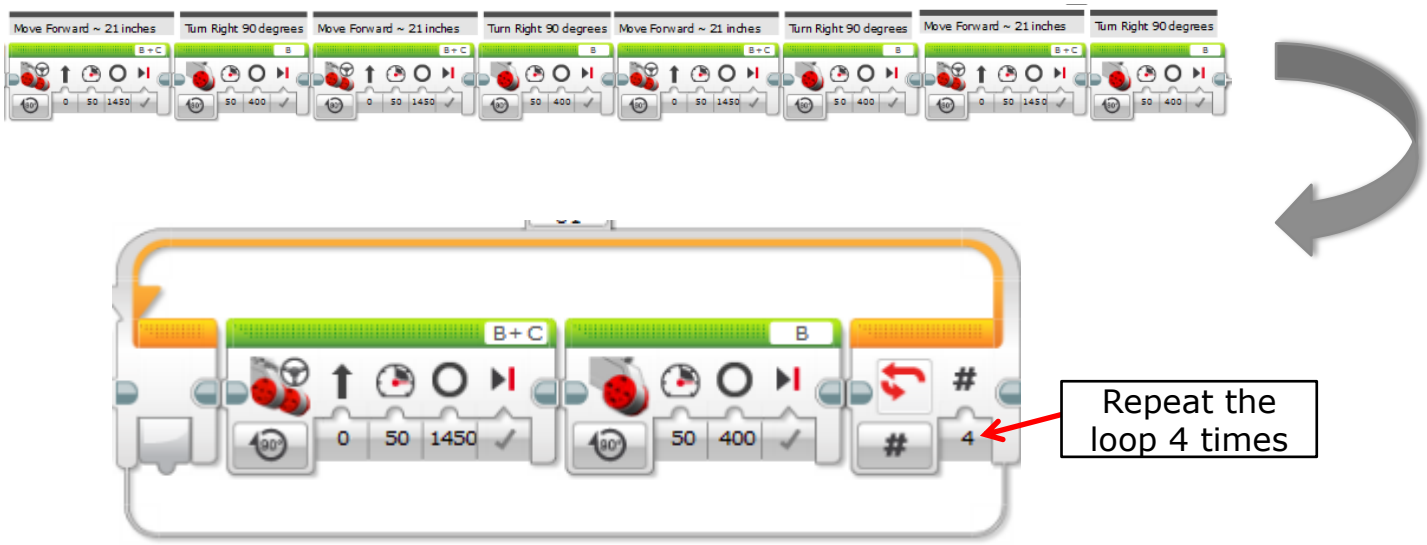
Is there an easier way?

Hint:

LOOPS

Loops make repeating a task multiple times easy

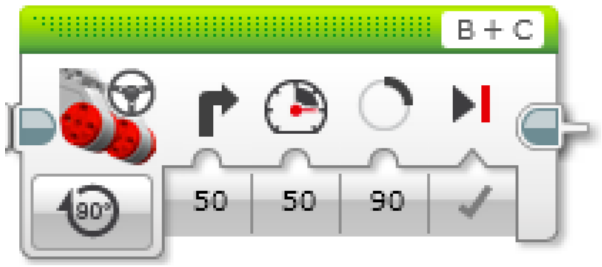
- KEEP GOING….Forever, for a Count, Until touch (or something else)



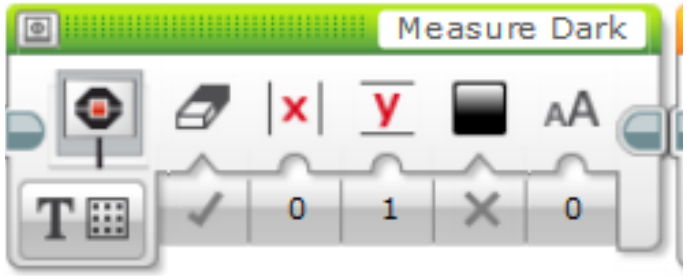
Challenge: Write a program to go around the box until touched

WHAT WE KNOW SO FAR

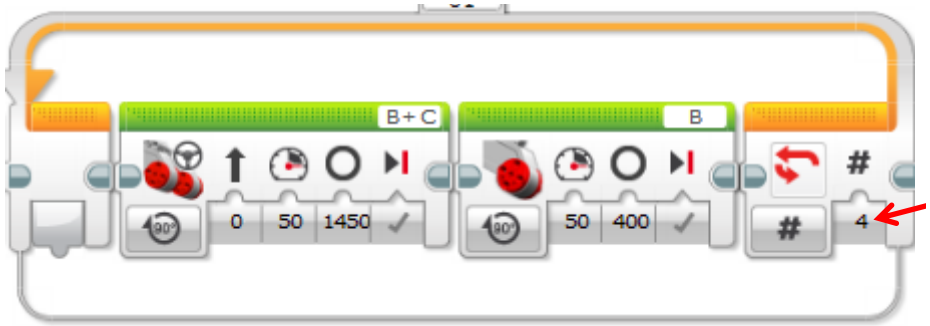
A



B



C



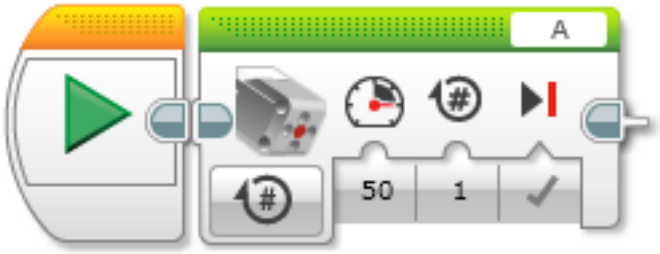
1) What is each Block called?

2) Where do you find it?

3) What does it do?

4) When do we use it?

D



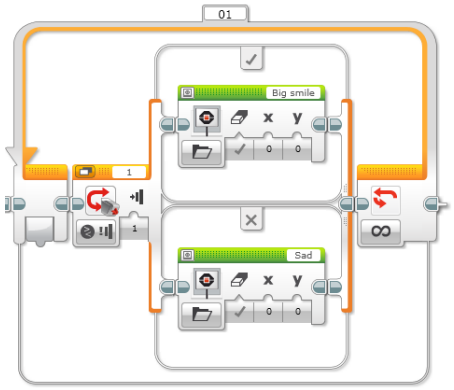
E

SECTION 7:
SWITCHING BETWEEN ACTIONS

SWITCH BLOCKS

Switch block

- Asking the robot a question and doing something different based on the answer
  - Example: Is the robot's touch sensor pressed? Or not?
  - Does the robot see a line? Or not?



- Basically a YES/NO QUESTION
- Switch blocks are found in the orange/flow tab

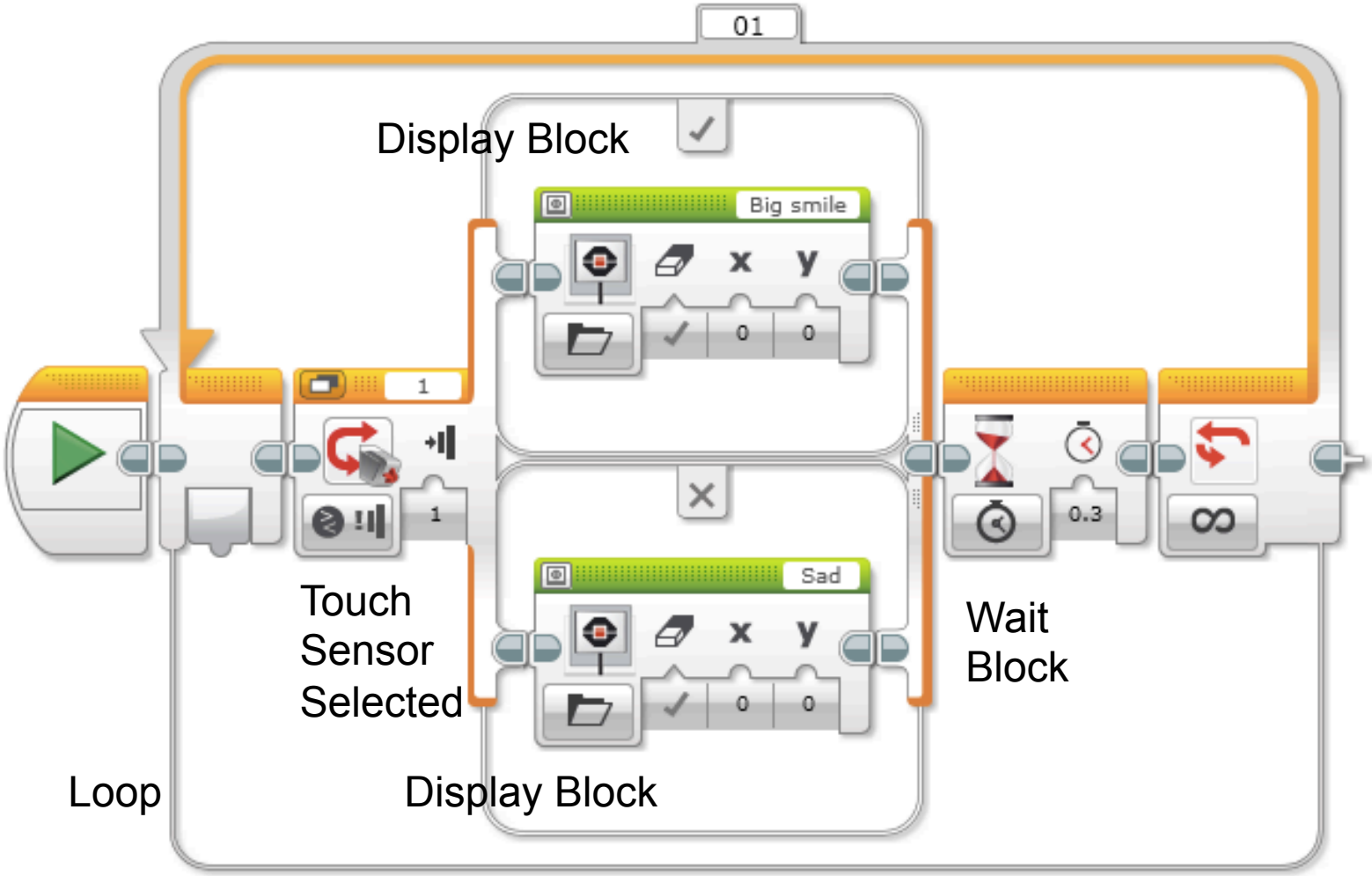
SWITCH BLOCK CHALLENGE 1

Challenge: **Write a program that changes based on if the touch sensor is pressed or not pressed.**

If pressed, your EV3 is happy! Display a smiley face. If not pressed, the EV3 is sad! Display a sad face.

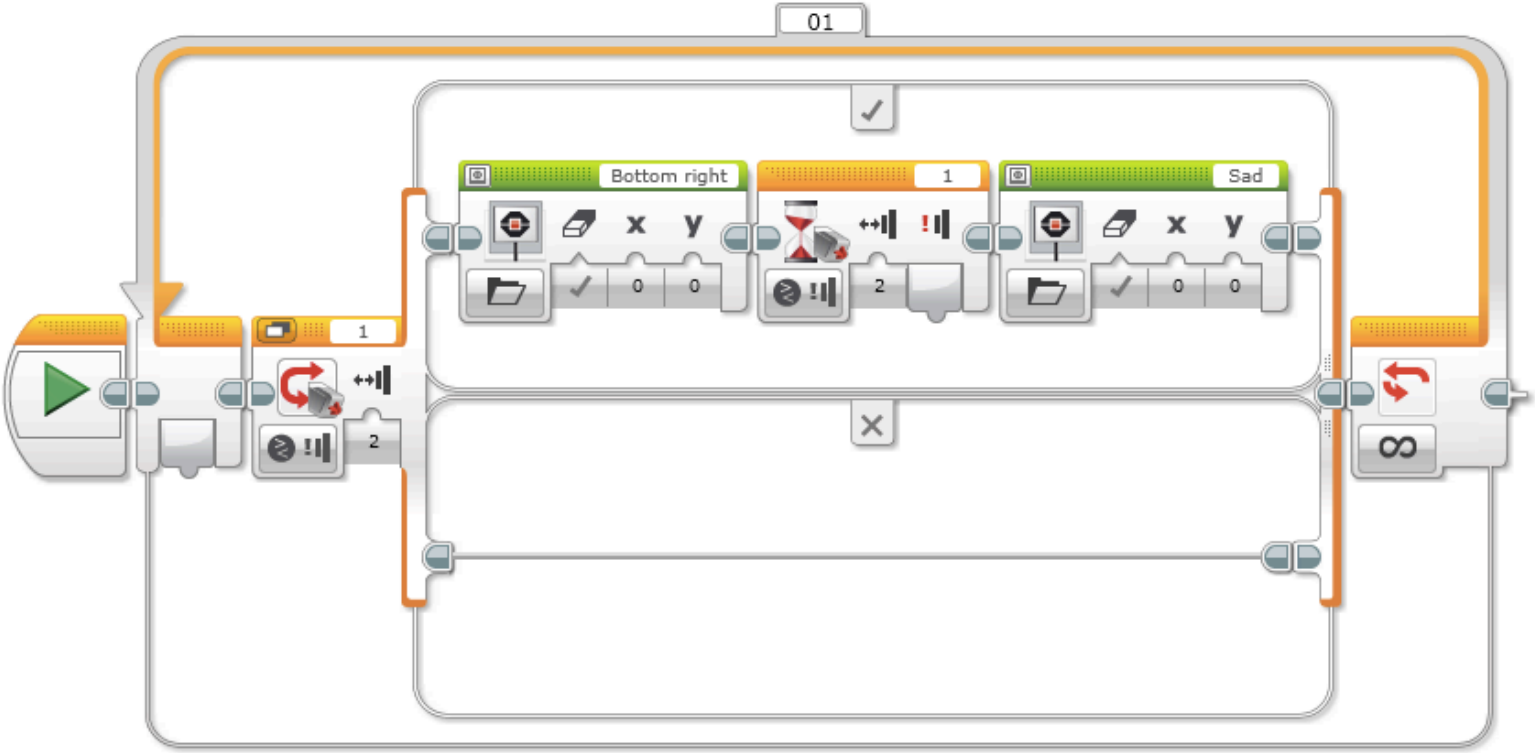
Hint: **You will need to use the display block and loops from yesterday and switch blocks from today!**

CHALLENGE 1 SOLUTION



SWITCH BLOCK CHALLENGE 2

Can you write a program that displays a smiley if you touch it once and a sad face if you touch it a second time and toggles back and forth.



SECTION 8: ULTRASONIC

MOVE FOR ULTRASONIC

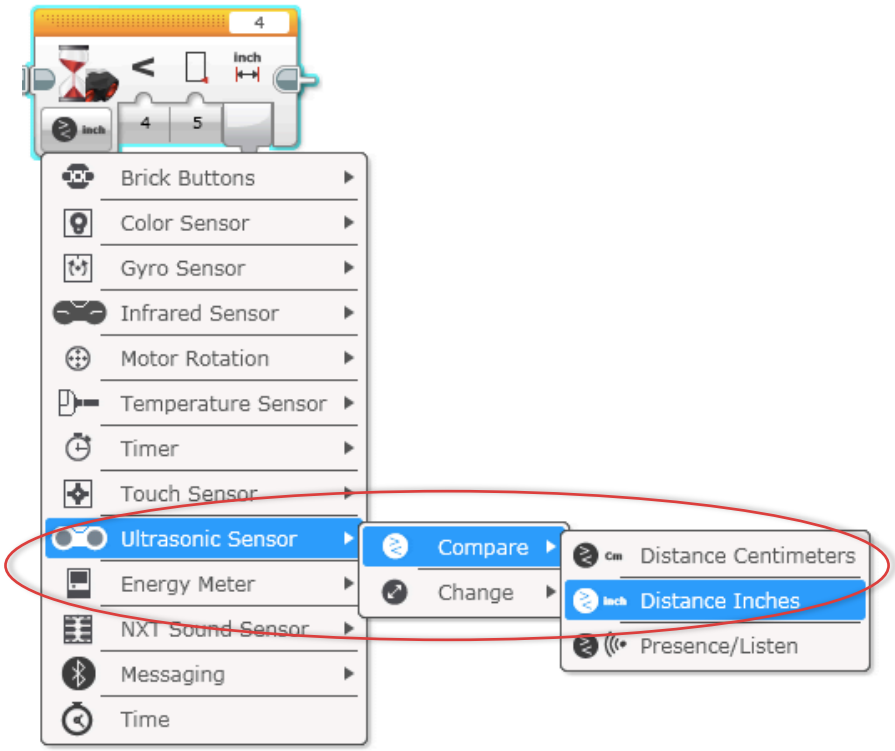
Challenge: Make the robot move until it is 5 inches away from the wall.

Step 1: Make a new program

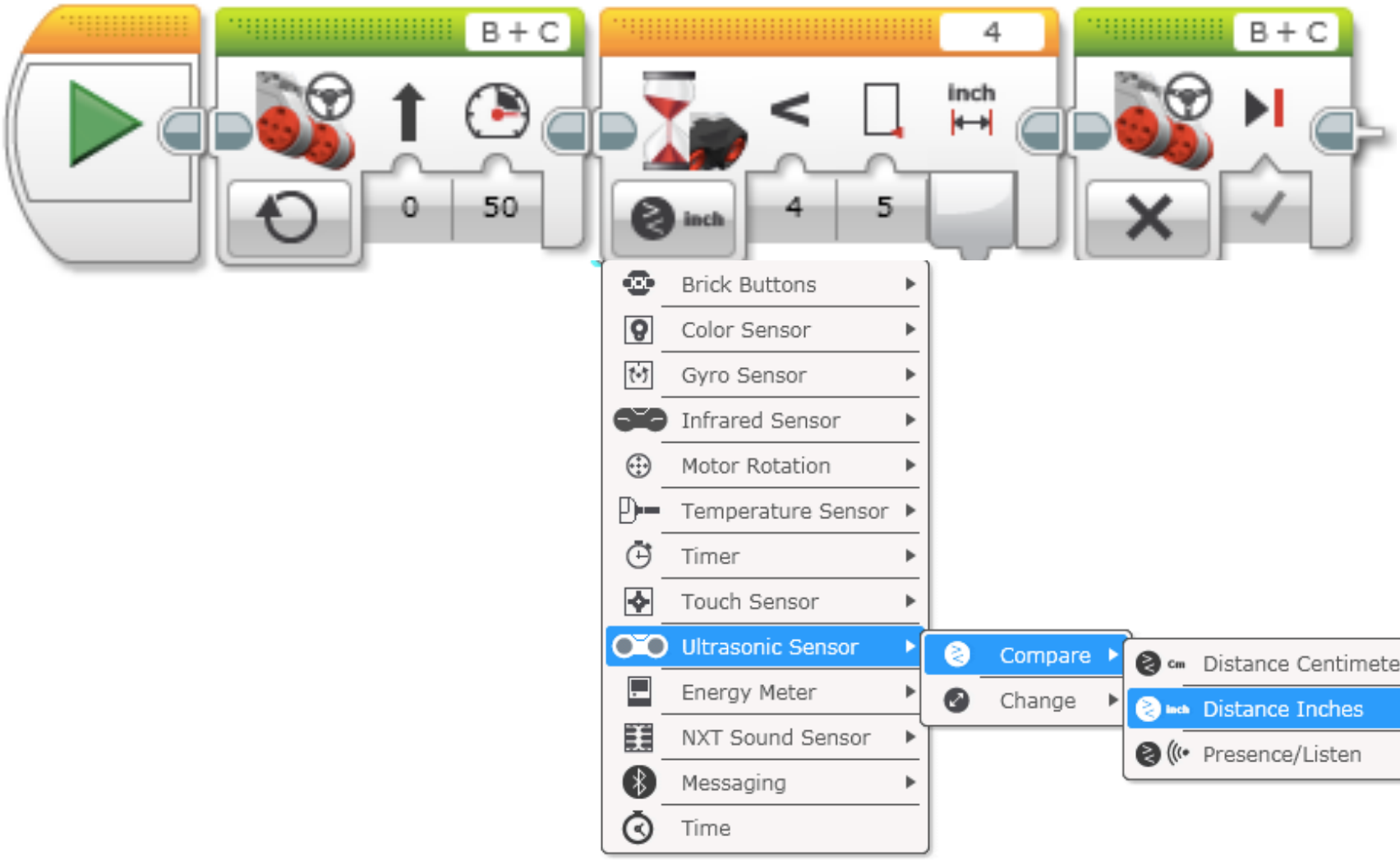
Step 2: Set move to “on”

Step 3: Set wait block to use the Ultrasonic

Step 4: Set move block to “off”



CHALLENGE 1 SOLUTION:



CHALLENGE 2: DOG FOLLOWER

Challenge: If the robot is closer than 5 inches away from your hand move backward, otherwise move forward.

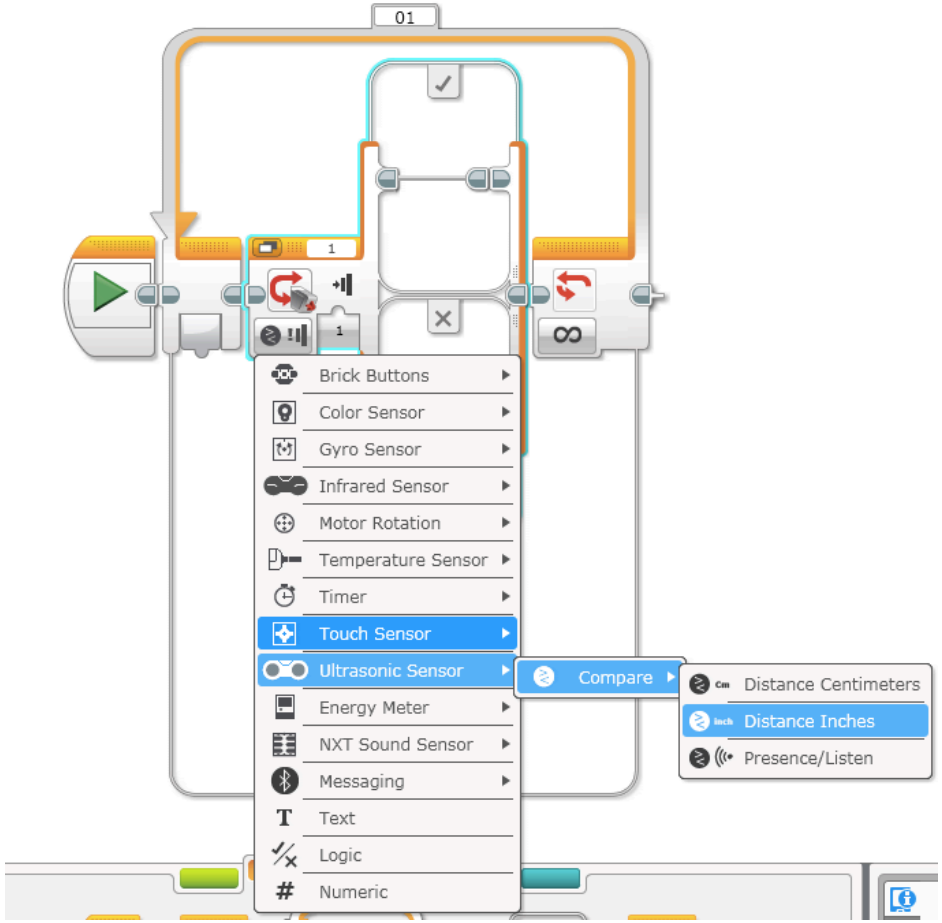
Step 1: Drag a loop from the orange tab

Step 2: Drag switch inside loop

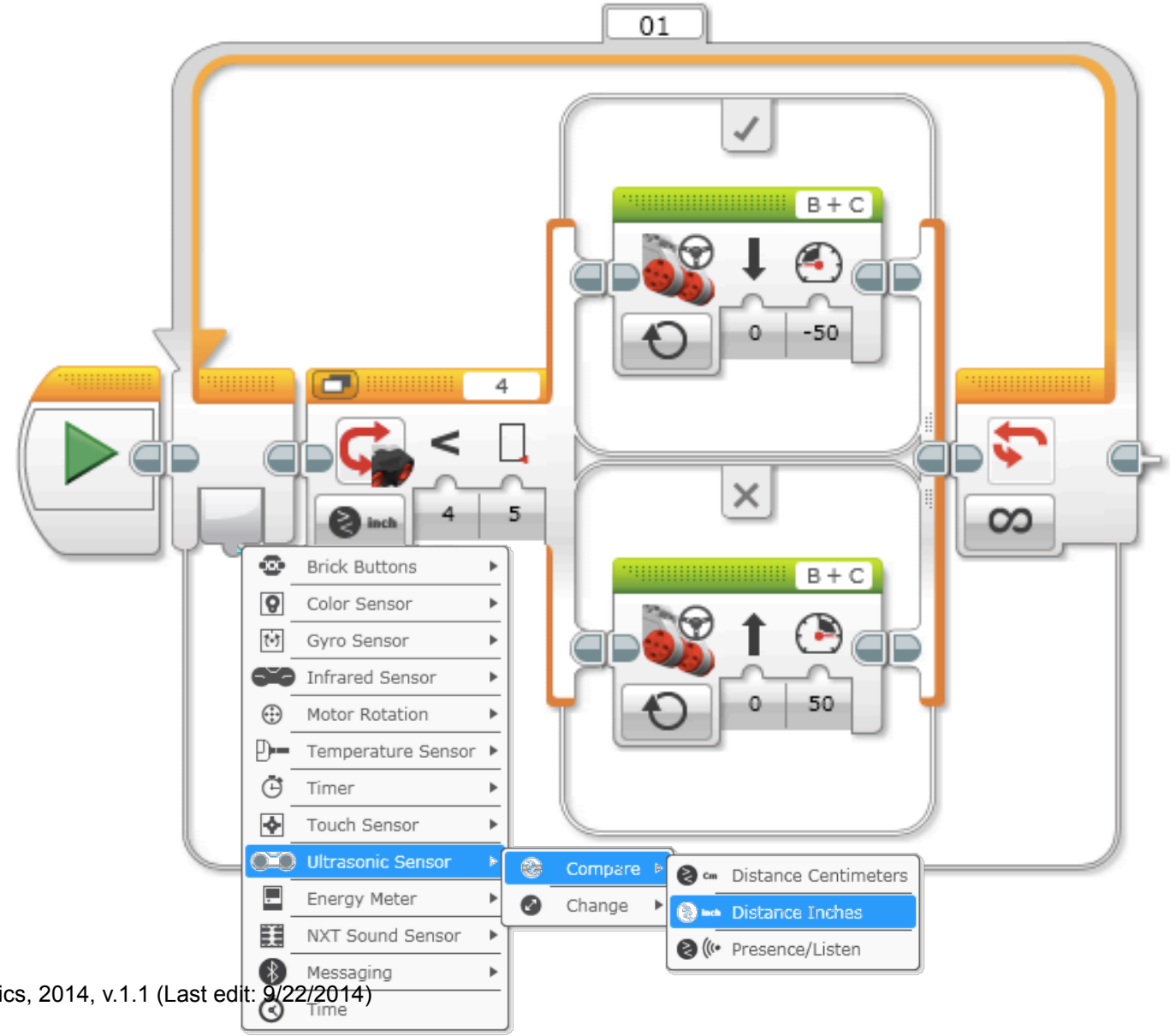
Step 3: Set switch to Ultrasonic

Step 4: Set move steering block to ON and place in TRUE

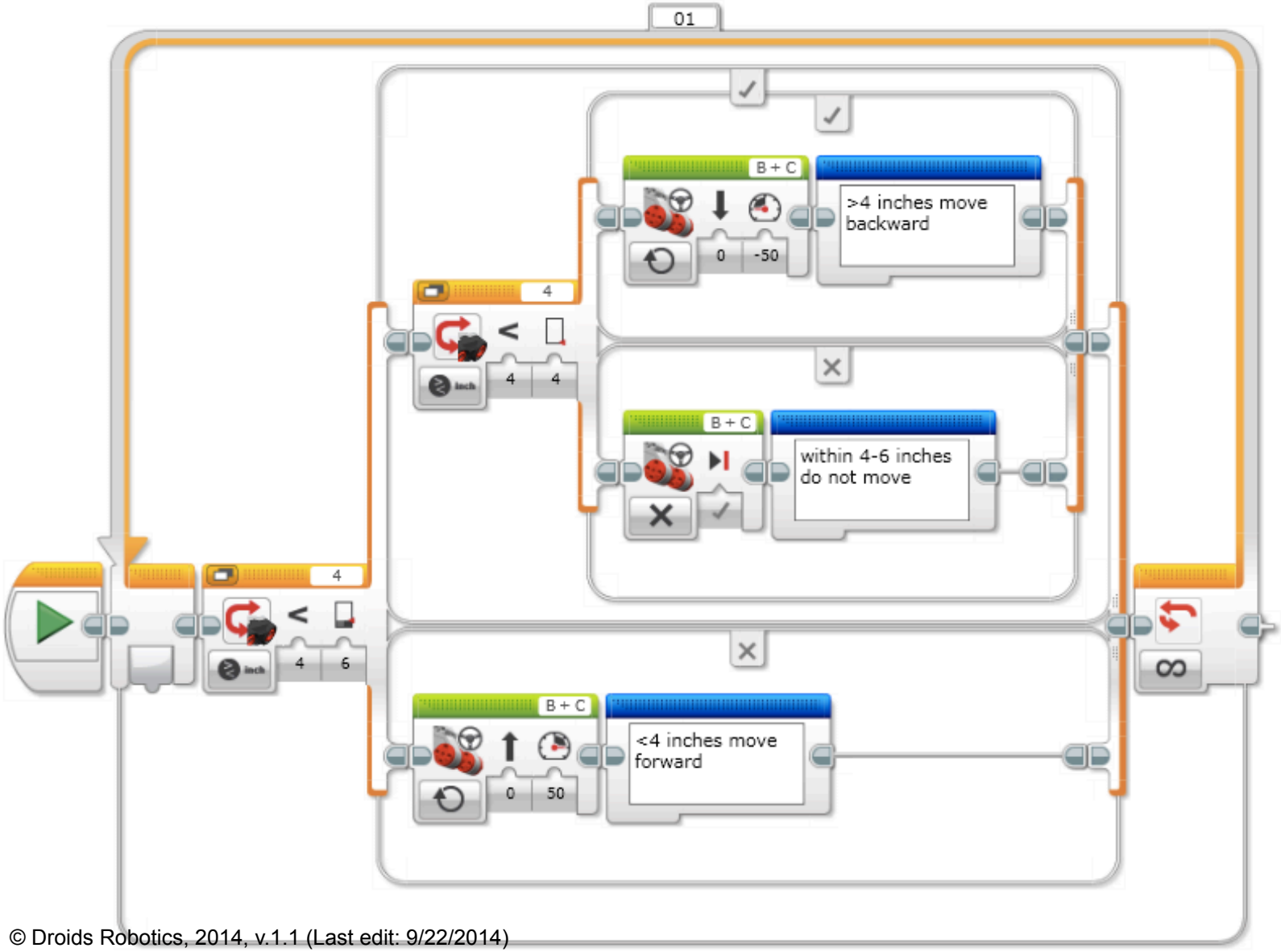
Step 4: Set move steering block to OFF and place in FALSE



CHALLENGE 2 SOLUTION:

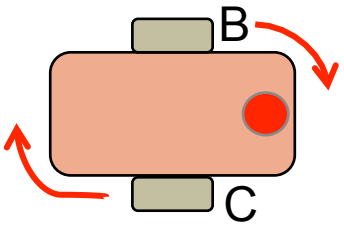


BETTER DOG FOLLOWER:

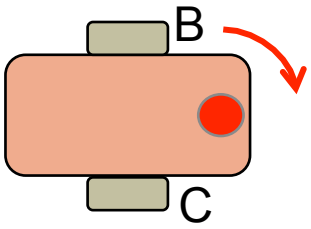


REVIEW

Turns: Spin and Pivot



Spin (used in tight spaces)



Pivot

Port View: Sensor readings on the brick

Wait Block:

Wait for a sensor reading (touch, ultrasonic, color) or time

Move until line, Move until touch

SECTION 9: LINE FOLLOWER

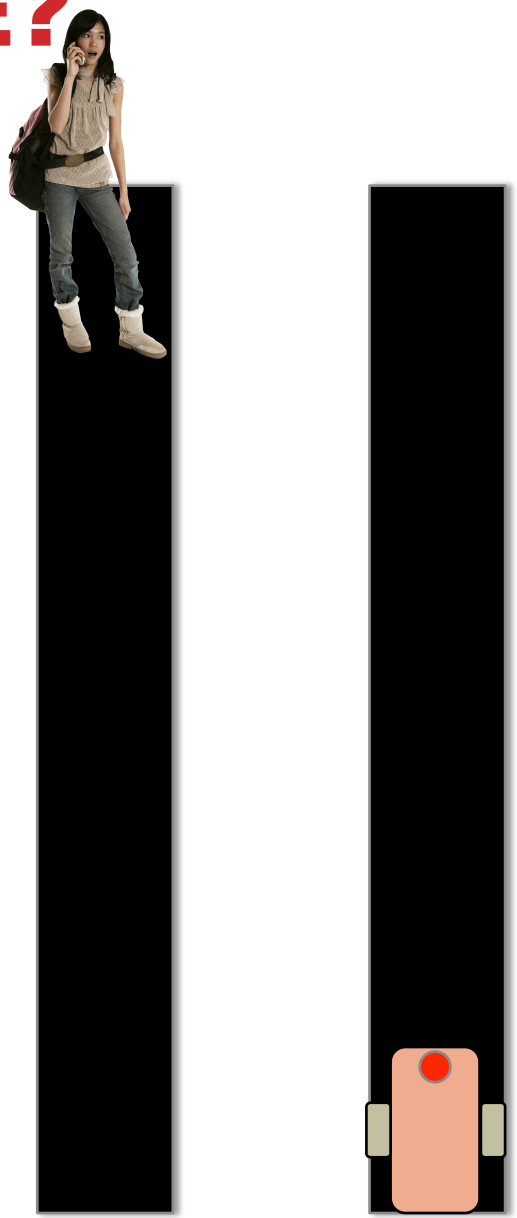
FOLLOW THE MIDDLE?

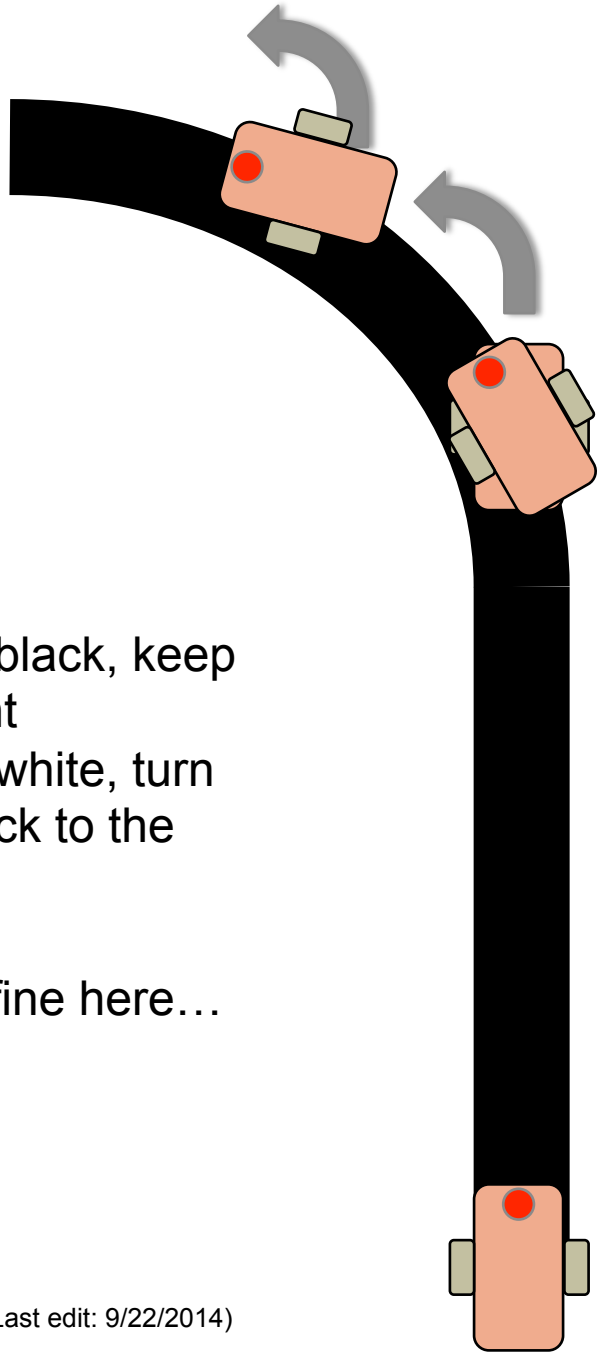
Humans want to follow the line in the middle.

Let's have the robot do the same thing using the Color Sensor

What type of questions can we ask using this sensor

- Are you on line or not?





1. If we are on black, keep going straight
2. If we are on white, turn left to get back to the line

Seems to work fine here…

1. If we are on black, keep going straight
2. If we are on white, turn left to get back to the line

OH NO… my robot is running away….

When the robot leaves the left side of the line, the program no longer works!

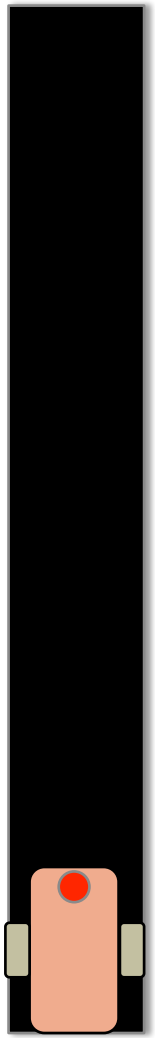
LINE FOLLOWING: ROBOT STYLE

Why could the Human follow the middle?:

- They can see ahead.
- They can see the whole line and its surroundings
- They see both sides and which side they left

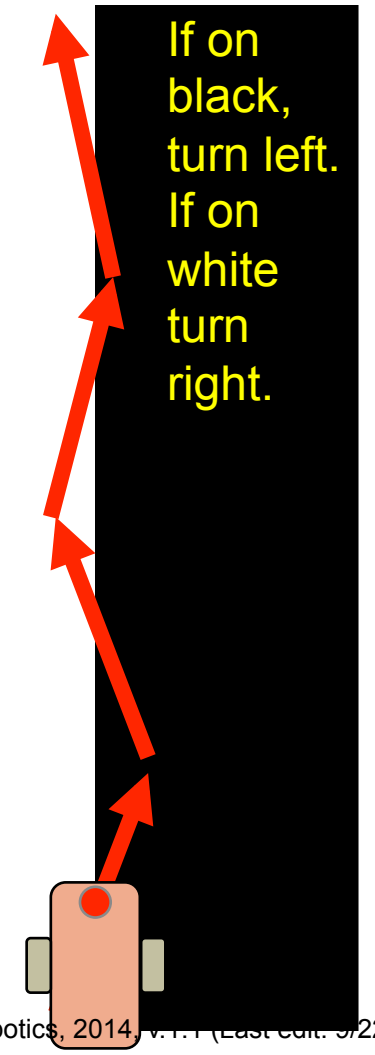
Why can't the Robot do the same thing?:

- Can't tell right or left side of the line
- **How do we make sure the robot always veers off on the SAME SIDE of the line?**
    - Instead of the middle, could the robot follow the "edge"?
- So now the robot will fall off only the same side.
- We will now show you how this works!



ROBOT LINE FOLLOWING HAPPENS ON THE EDGES

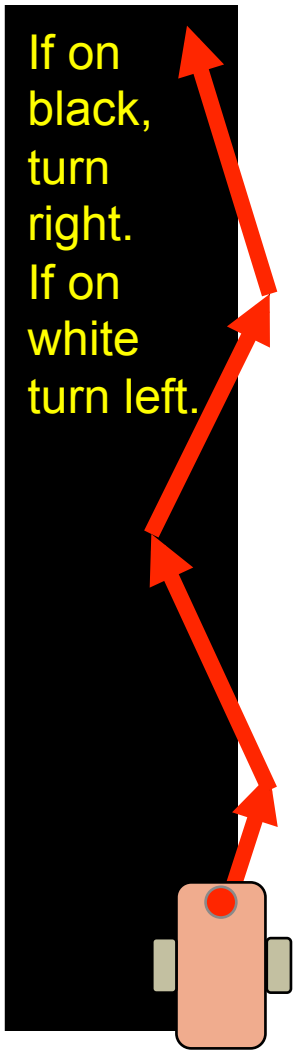
Left side line following



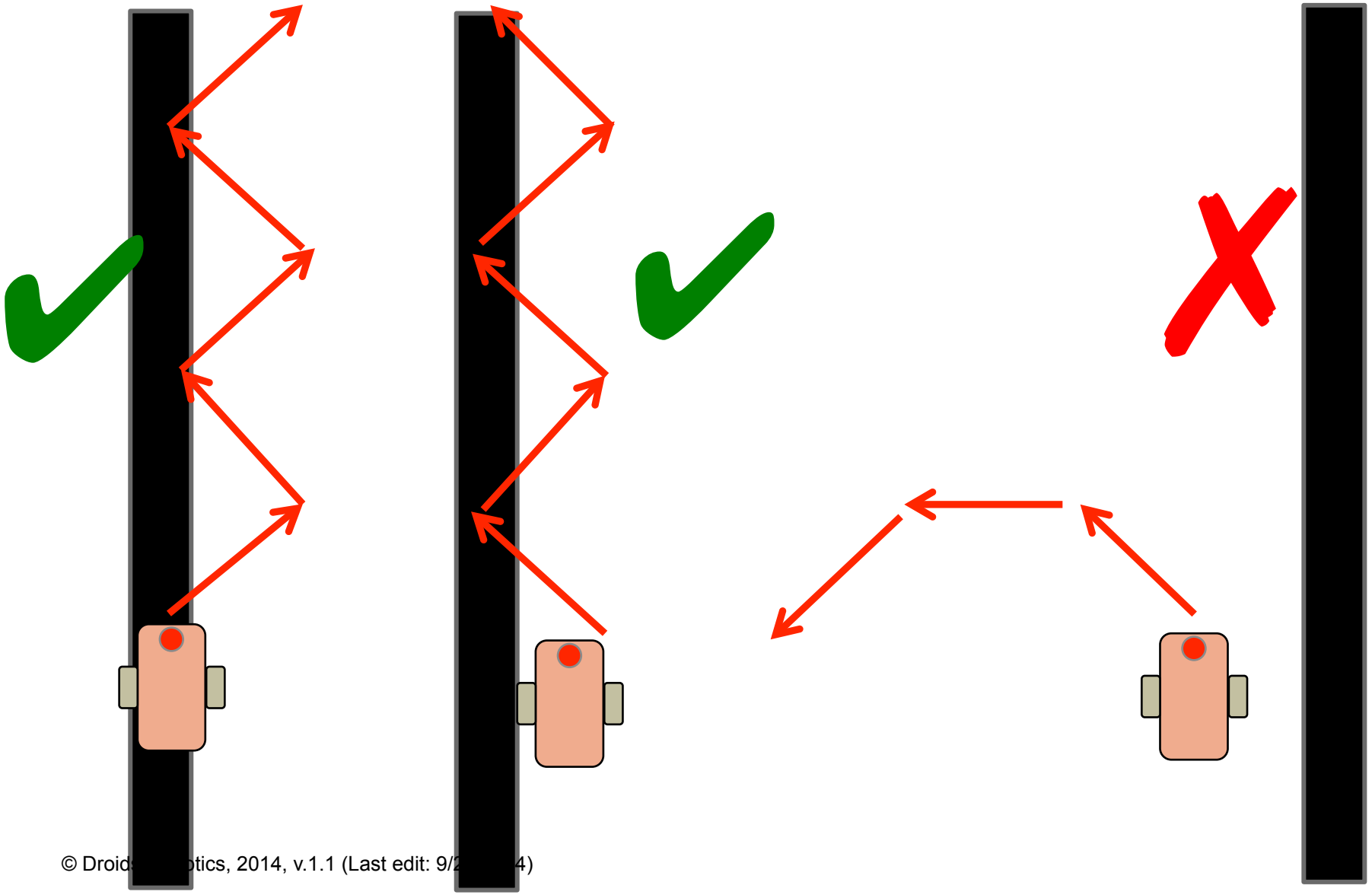
The robot has to choose which way to turn when the color sensor sees a different color.

The answer depends on what side of the line you are following!

Right side line following



STARTING THE ROBOT ON THE CORRECT SIDE



LINE FOLLOWER CHALLENGE

Step 1: Write a program that follows the RIGHT edge of a line.

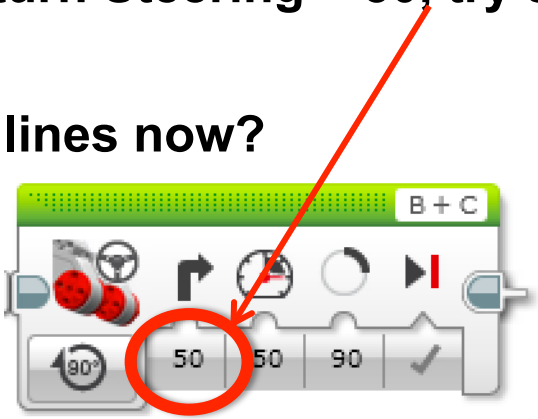
Hints: If your sensor sees black, turn left. If your sensor sees white, turn right. Use loops and switches!

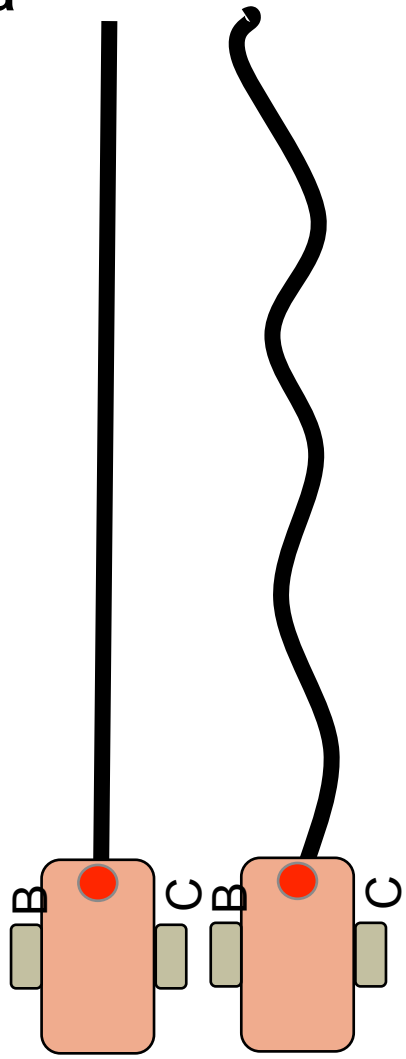
Step 2: Try it out on different lines.

Did your line follower work the same on straight and curved lines?

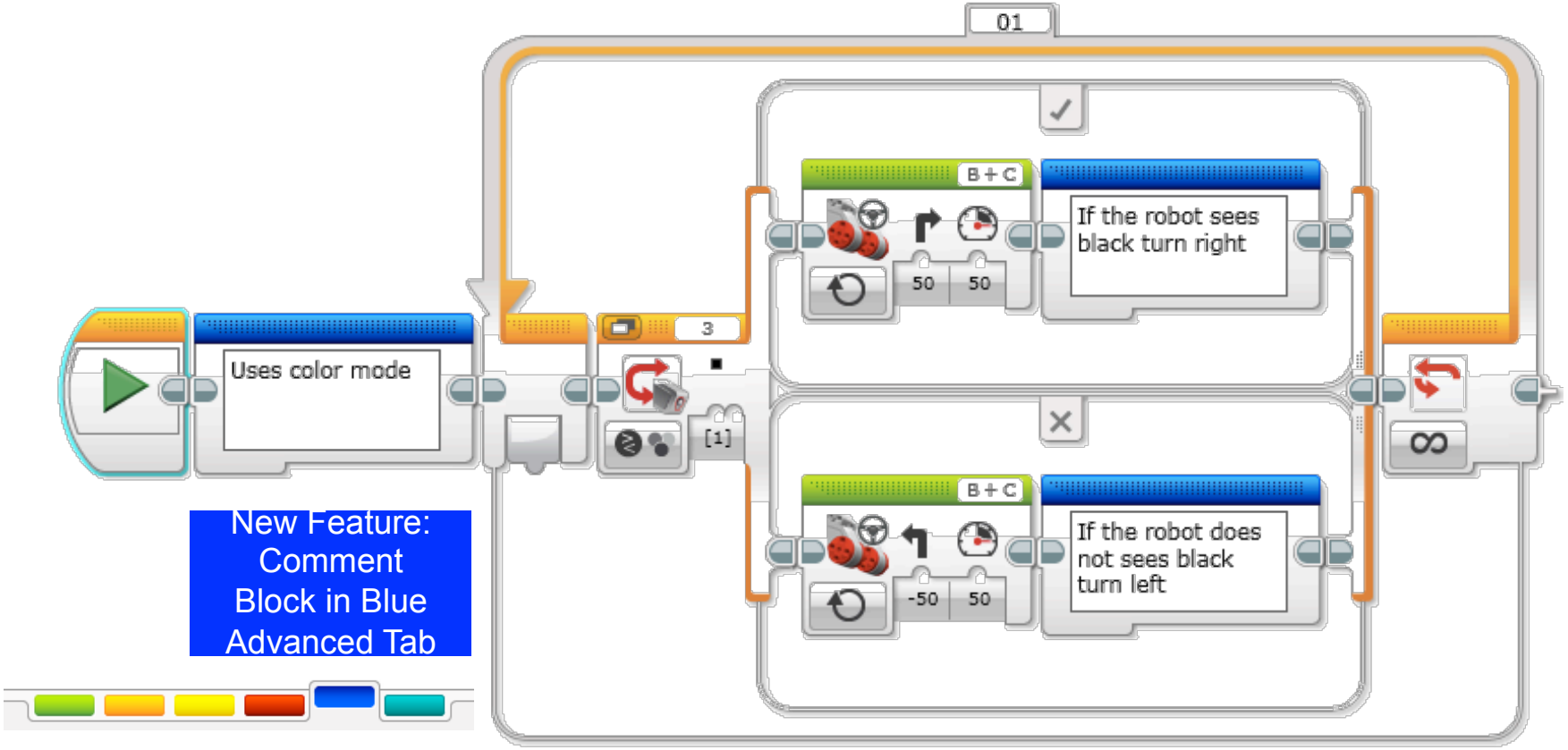
Step 3: If not, instead of turn Steering = 50, try smaller values.

Is it better on the curved lines now?



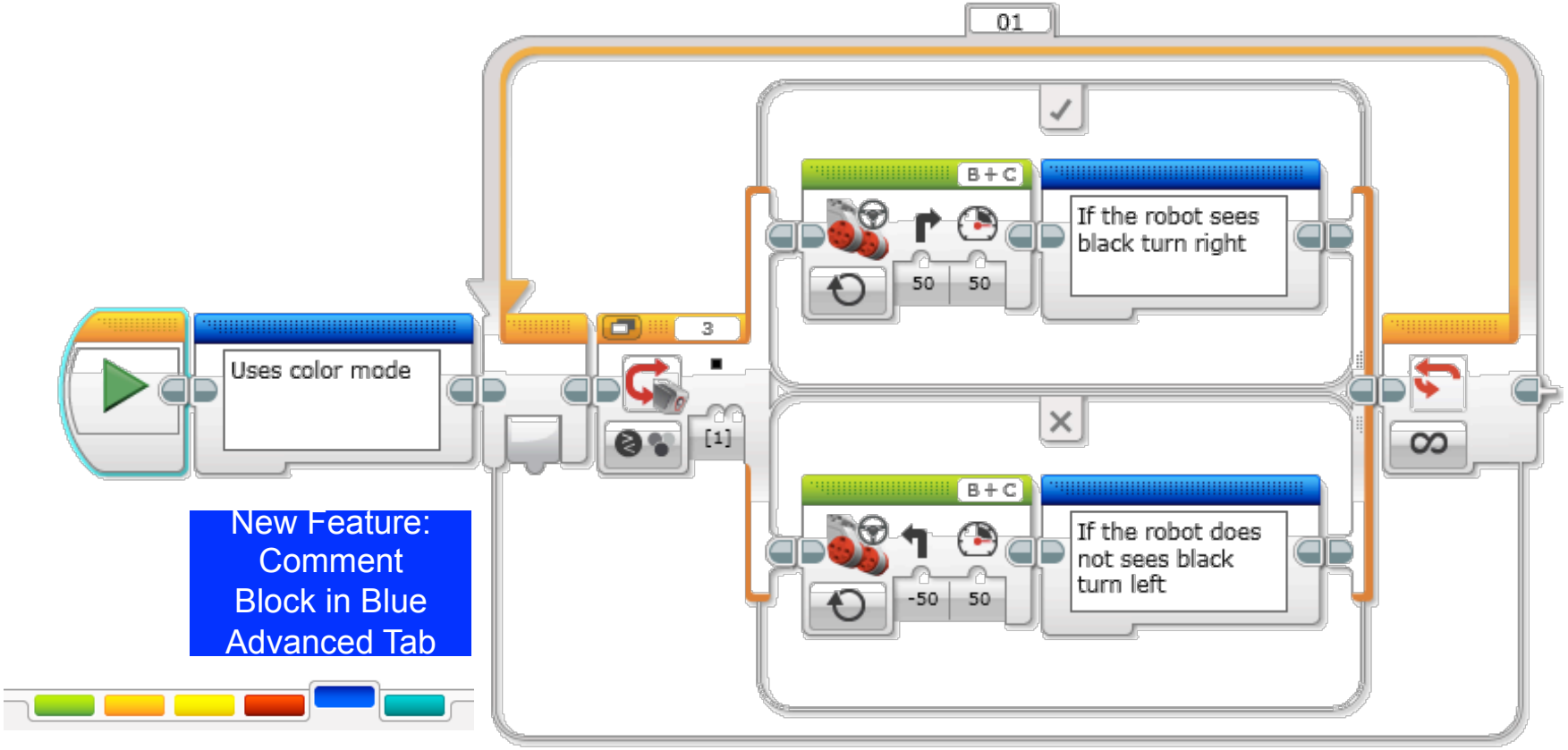


LINE FOLLOWING CHALLENGE SOLUTION



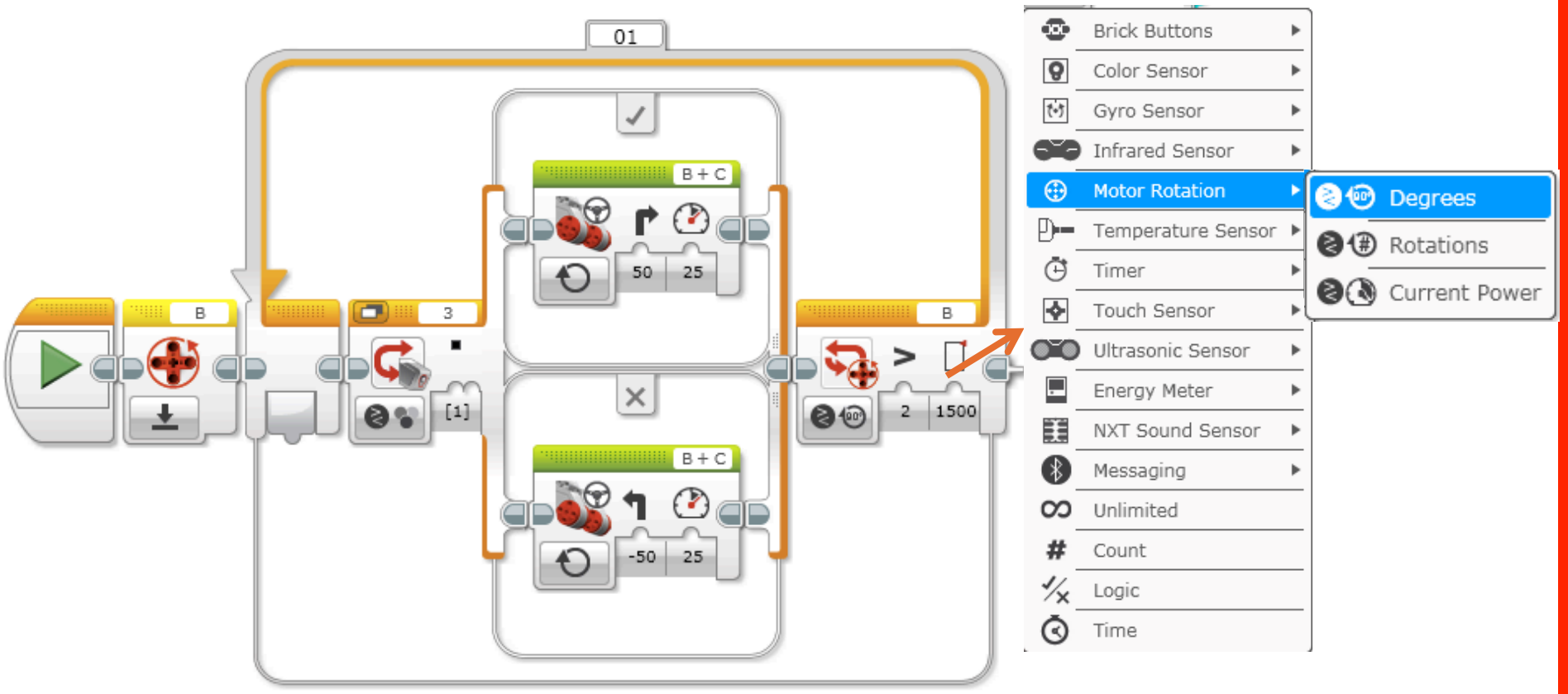
Does this program follow the Right or Left side of a line?

LINE FOLLOWING CHALLENGE SOLUTION

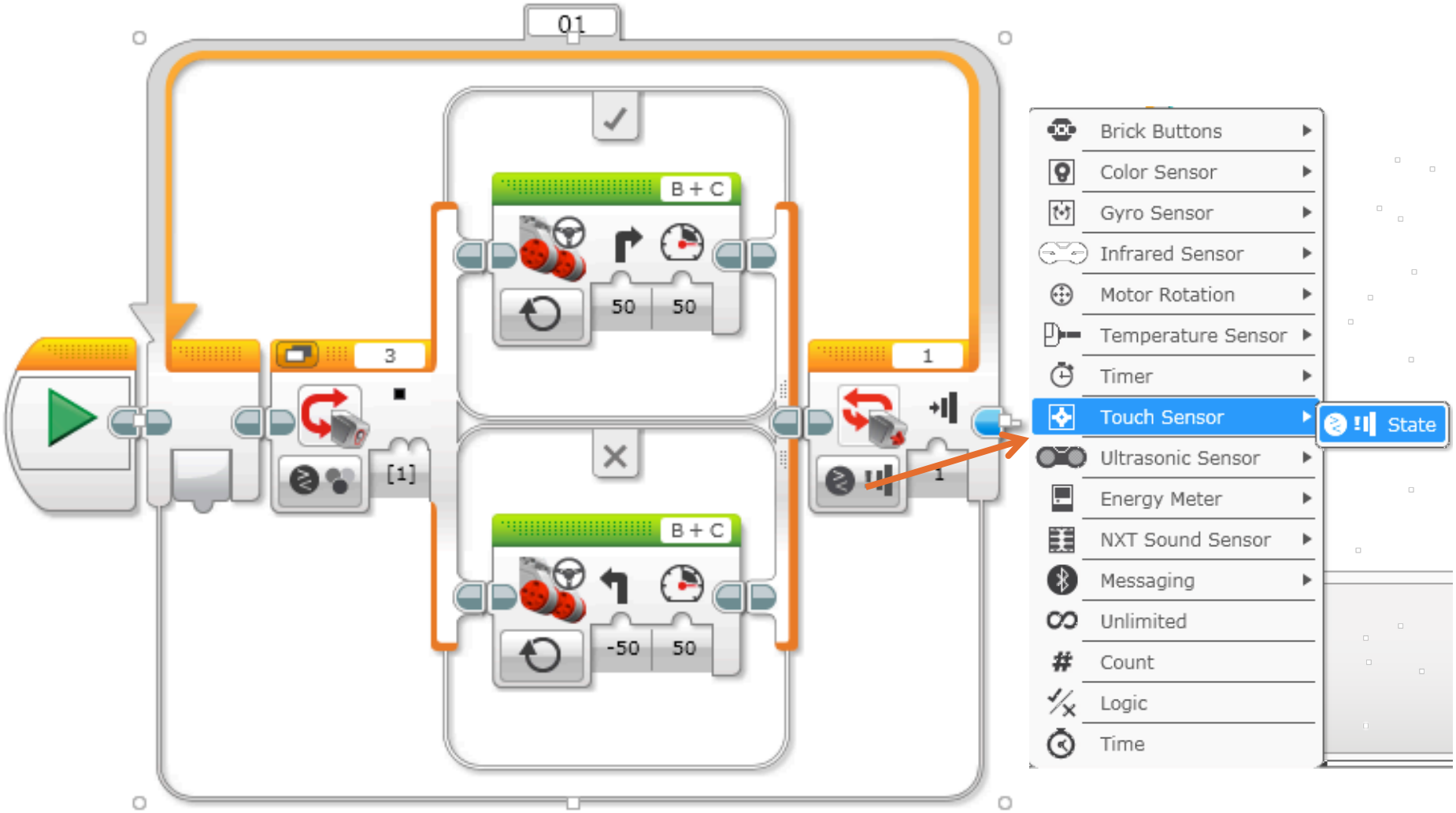


How do we make this stop?

LINE FOLLOWING FOR A SENSOR OR DISTANCE



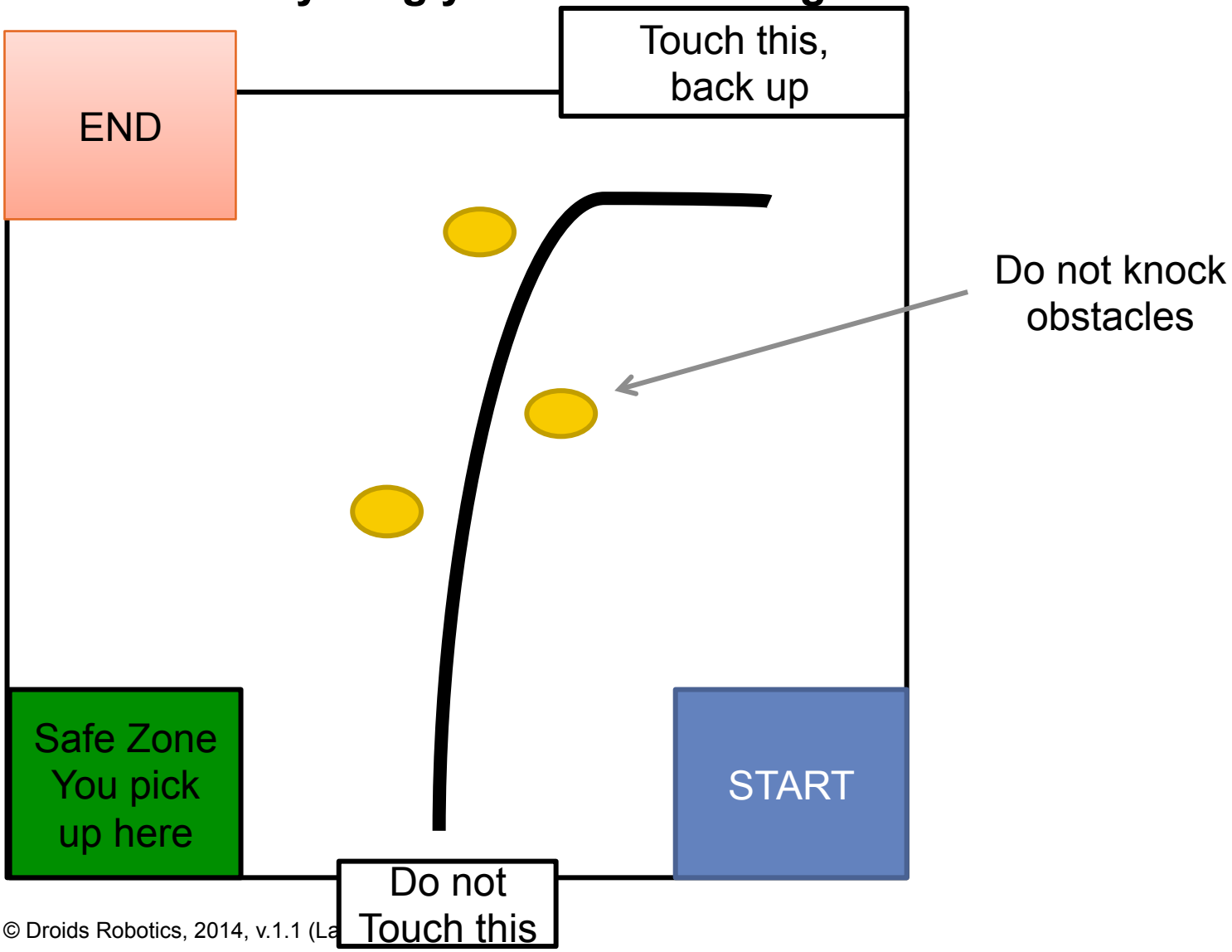
LINE FOLLOWING FOR A SENSOR OR DISTANCE



SECTION 10: FINAL CHALLENGE

FINAL MAZE CHALLENGE

Combine everything you know to navigate a maze like this.



CREDITS

- This tutorial was created by Sanjay Seshan and Arvind Seshan from FLL Team Not the Droids You Are Looking For (Droids Robotics)
- We have additional material for more advanced lessons available on request.
- Useful tools for FLL teams and robot programmers are available at www.ev3lessons.com
- The material is made available to you free of charge. However, we would greatly appreciate a letter indicating that you are using the materials and what you think of them.
- Feedback and suggestions are encouraged.
- Email: team@droidsrobotics.org