Vector Navigation

Programming and Operating Instructions

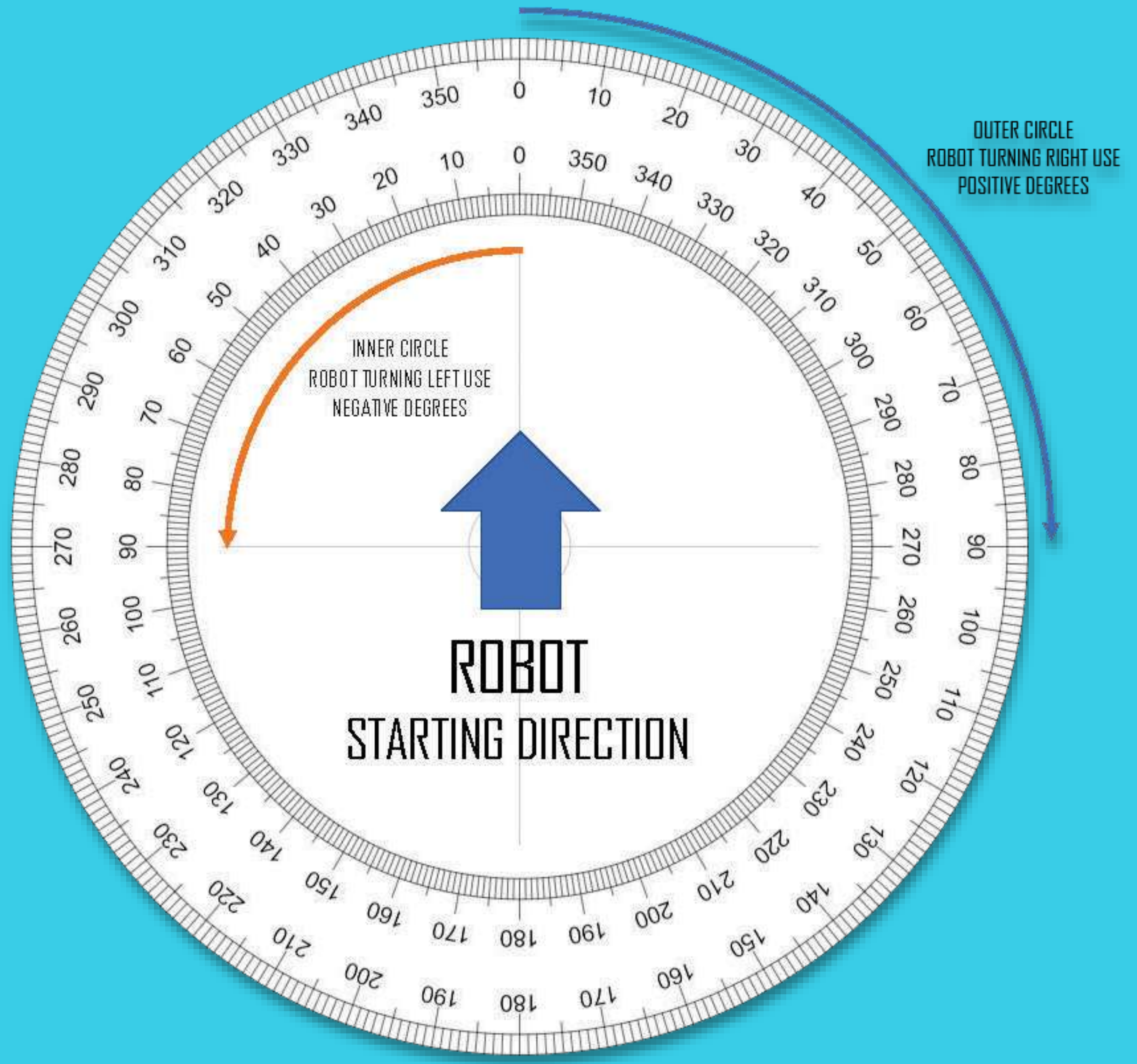
2019

Vector Navigation Description

- A vector has velocity (speed and direction) and magnitude (length).
- Navigation is the movement from one location to another.
- Vector Navigation is a LEGO® Mindstorms® My Block that provides a Mindstorms® robot speed, direction, and length to travel.
- With four My Blocks a robot can navigate around the FIRST® LEGO® League (FLL®) Challenge Table.
  - The Gyro Calibrate (GyroCal) My Block to counter gyro drift.
  - The Vector Move (VecMove) My Block to navigate.
  - The Pin Turn - Right (PinRight) and Pin Turn - Left (PinLeft) My Blocks are used only if changing direction and traveling a short distance. They function independent from the Vector Move My Block and the compass.

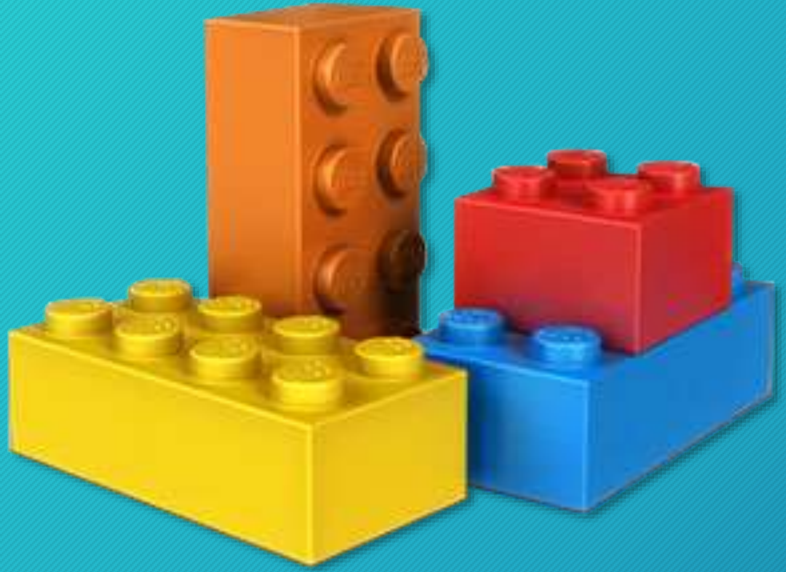
Vector Compass

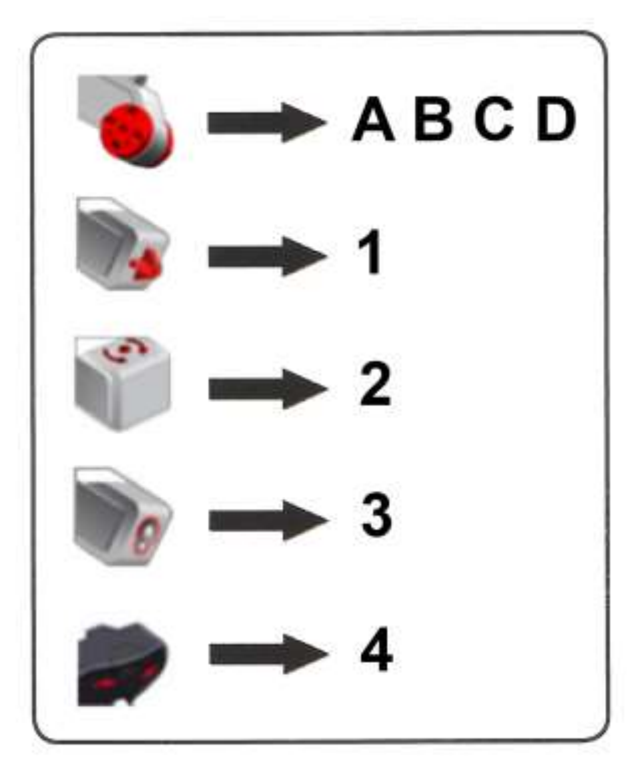
- Shows the direction (angle) to travel.
- Zero is the direction the robot starts at, and stays the same throughout the run.
- Positive degrees rotate the robot right (outer circle) and negative rotates it left (inner circle).
  - Keep in mind where the robot is. For example, if the robot is facing -90 and you want to go 180, entering 180 will turn the robot right past 0 until it reaches 180. Entering -180 will turn the robot left until it reaches -180.



My Blocks Notes

The My Blocks in these examples use the EV3 default ports for sensors and motors. Please adjust the port settings if you are not using the defaults.





EV3 Default Ports

Preliminary setup

Things to do before starting Vector Navigation programming

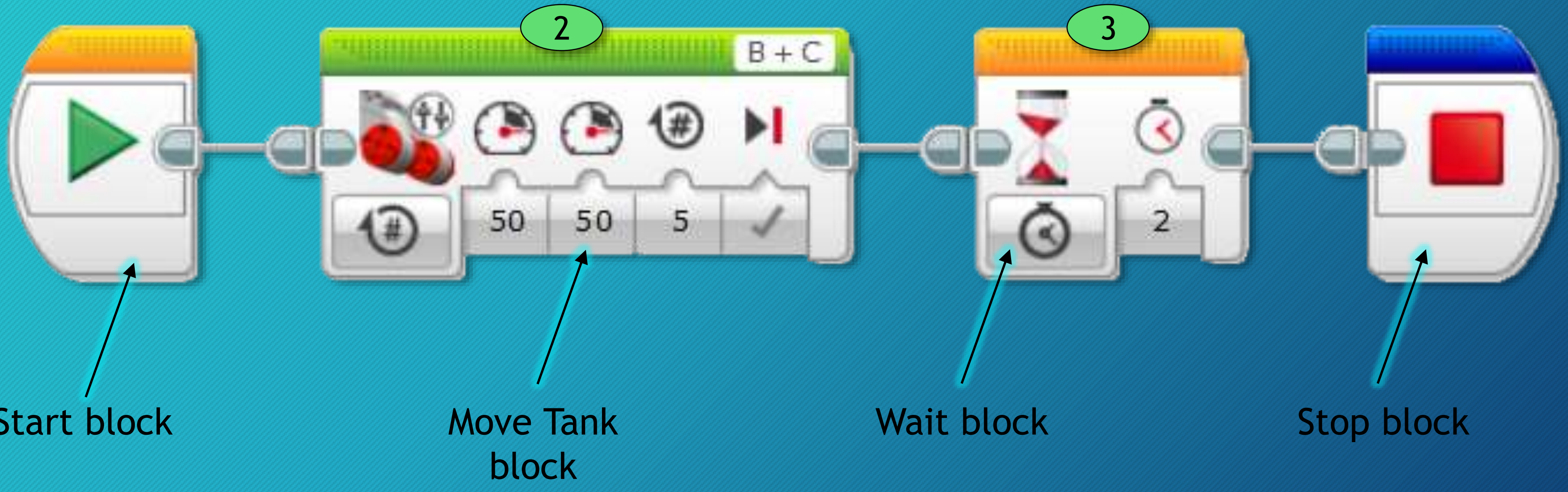
Determine wheel circumference

The Vector Navigation My Block uses the circumference of the robot's drive wheels.

- There are several ways to determine the circumference:
  - Calculate from what is printed on the wheel.
  - Measure wheel diameter.
  - Use robot to determine circumference (best!).

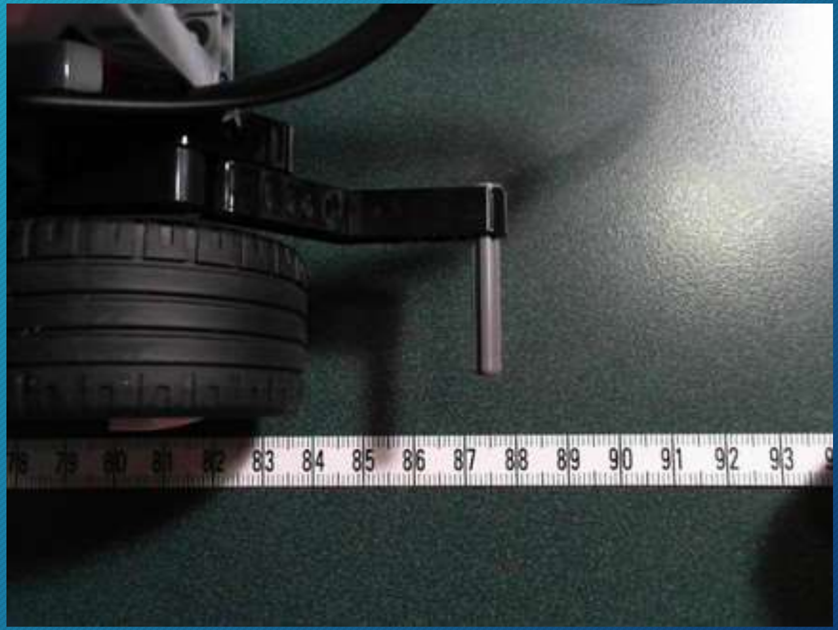
Calculate wheel circumference

1. Create a new program named Move5.
2. Insert a Move Tank block set to, 50 power and five (5) rotations.
3. Insert a Wait block set to 2 seconds.



Calculate the circumference

1. Use a reference on the robot to mark starting point (Axle in photograph).
2. Run the Move5 program.
3. Using a tape measure, measure distance traveled by robot in centimeters.
4. Record distance.
5. Repeat steps 1-4 five times.



TIP: Use centimeters for measuring units.

Calculating wheel circumference

Add the five distances traveled together:

87.3 + 87.6 + 87.9 + 87.0 +88.2 = 438

Average distance traveled:

438 ÷ 5 = 87.6

(Total of the test runs ÷ Number of test runs = Average of test runs)

This sample uses the stand LEGO® EV3 Education Set rim and tire.

Calculate the circumference using the robot

Distance traveled ÷ Motor rotations = Wheel circumference

OR

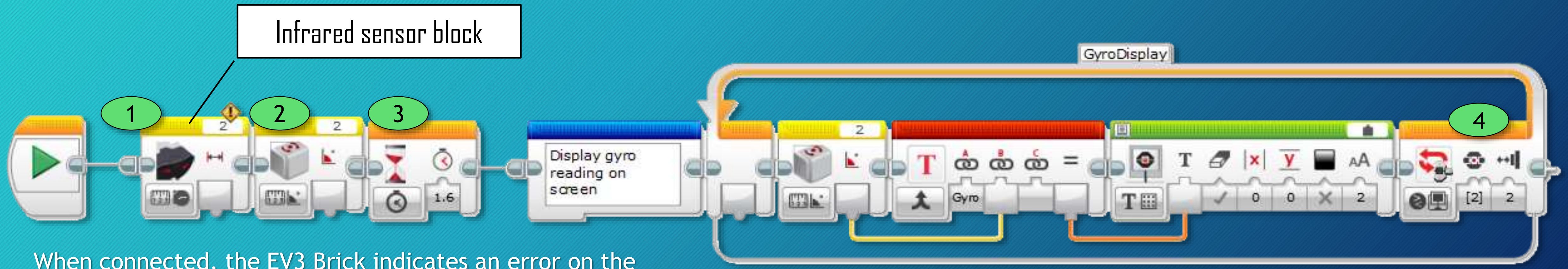
87.6 cm ÷ 5 = 17.52 cm

Gyro Calibrate

- Vector Navigation uses the Gyro. The gyro calibrate (GyroCal) My Block "calibrates" the gyro reducing drift.
- This version works with new and old version of gyros.
- The gyro calibrate needs ran once at the beginning.
- Robot **must** be stationary and on the surface it will be running on. It cannot be moving during calibration.
- The GyroCal My Block displays a real-time gyro reading on EV3 screen so you can check for drift. If still drifting (the displayed number changing without the robot moving) rerun the program.

Gyro Calibrate

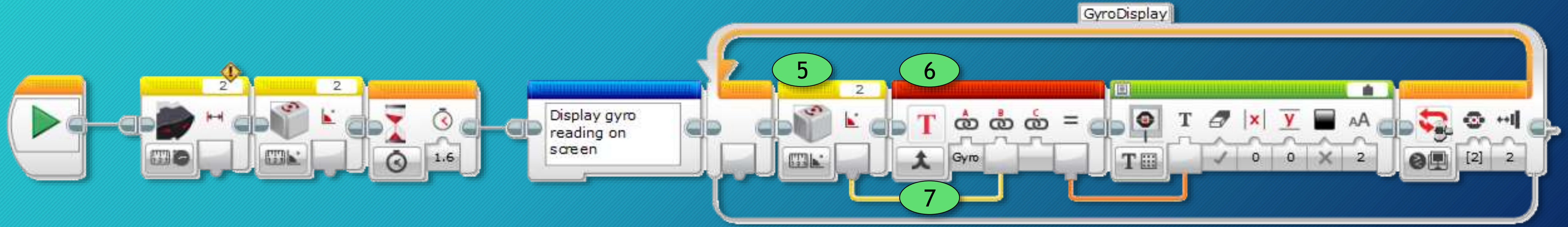
1. Insert an Infrared sensor block set to the same port as your Gyro (port 2).
2. Insert a Gyro sensor block set to port 2, and Measure → Angle.
3. Insert a Wait block set to Time and 1.6 second.
4. Insert a Loop block set to Brick Buttons → Compare, Brick Button ID: 2, and State: 2.



When connected, the EV3 Brick indicates an error on the Infrared Sensor block. This is correct and required for the program to work.

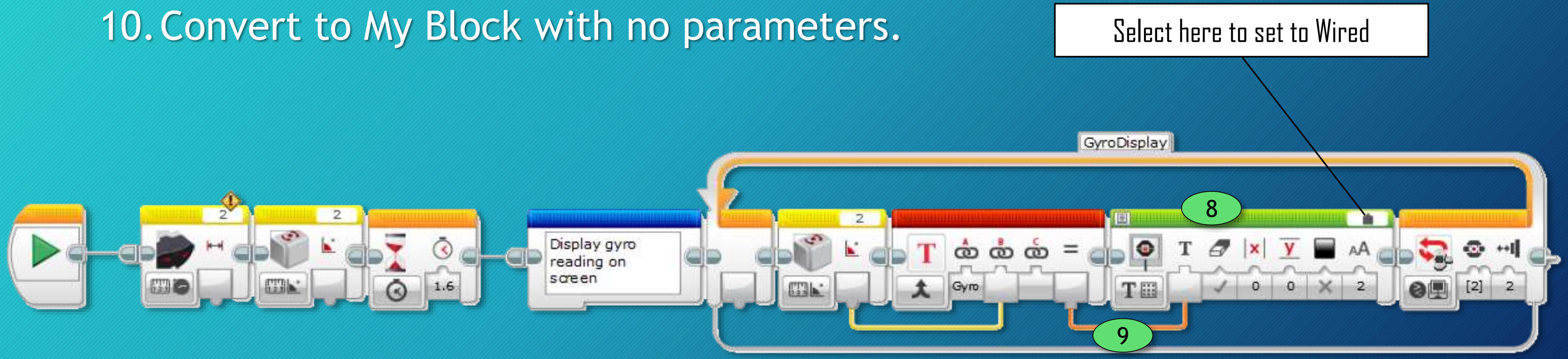
Gyro Calibrate

5. Inside the Loop block, insert a Gyro sensor block set to port 2 and Measure → Angle.
6. Insert a Text block. In A, type Gyro followed by a space.
7. Drag a wire from the Gyro sensor block and place it in B input of the Text block.



Gyro Calibrate

8. Insert a Display block set to Wired, Clear Screen: true, and Font: 2.
9. Connect a wire from the Text block output (=) to the Display Text input.
10. Convert to My Block with no parameters.

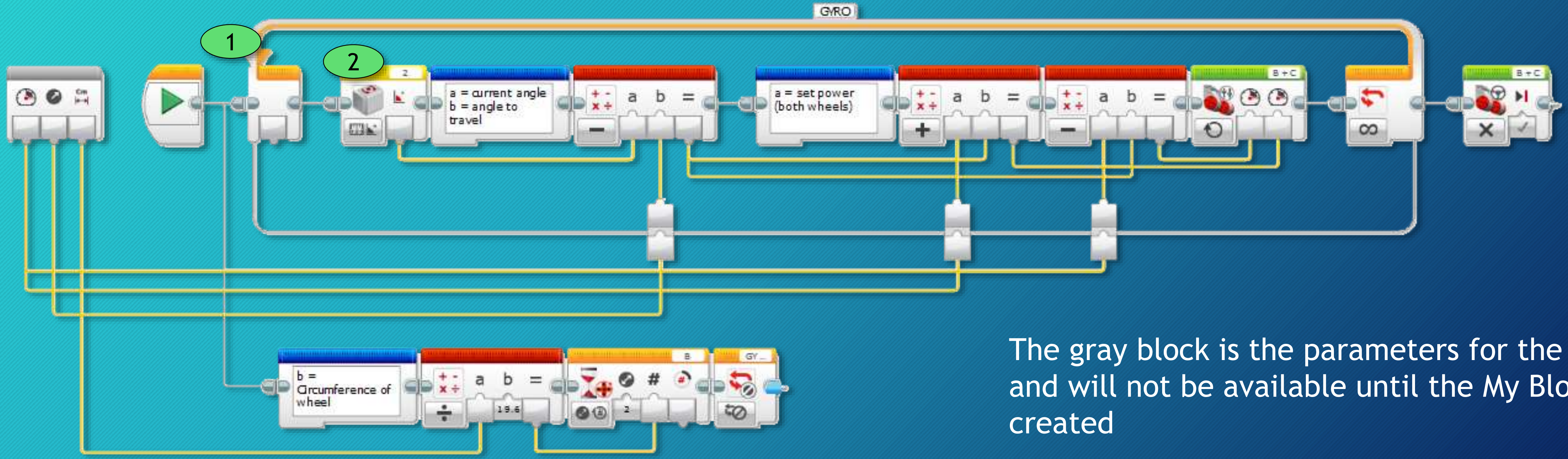


Vector Navigation

My Block programming instructions

Vector Navigation

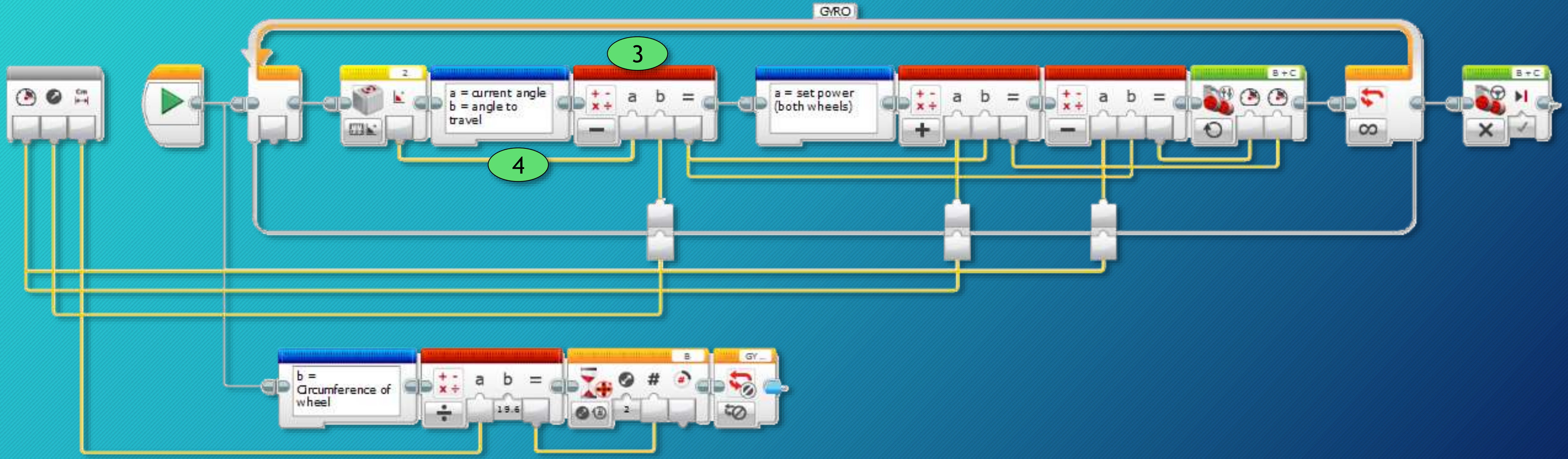
1. Insert a Loop block after the Start block.
2. Inside the Loop block, insert a Gyro sensor block set to Measure ➔ Angle and set to port 2.



The gray block is the parameters for the M and will not be available until the My Block created

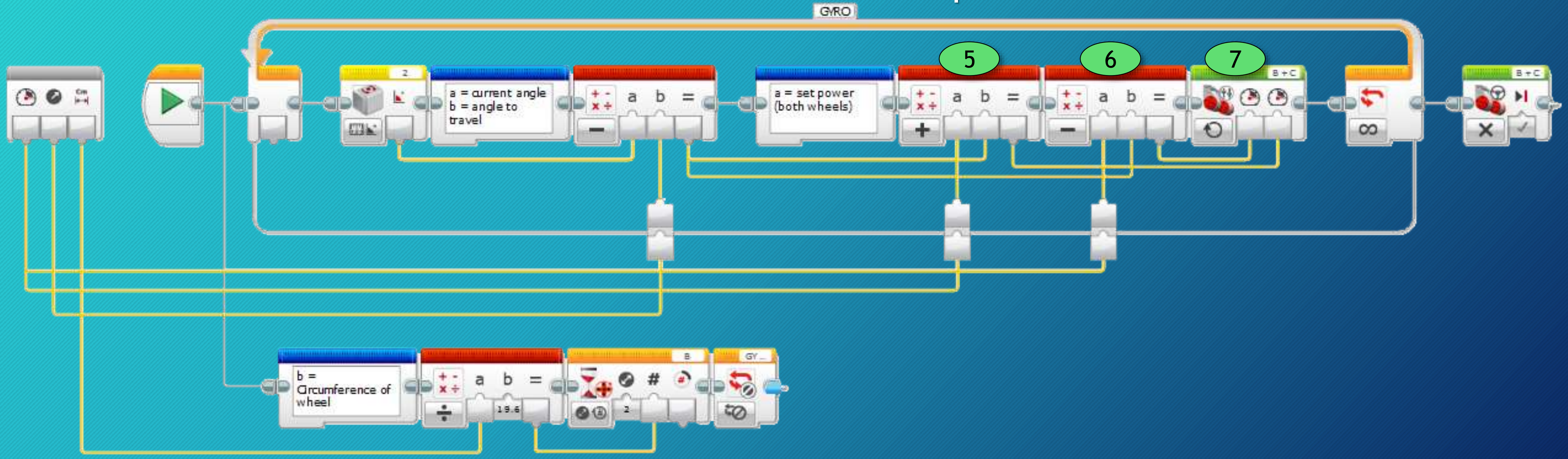
Vector Navigation

3. Insert a Math block set to subtract (-).
4. Connect a wire from the Gyro sensor block to the first Math block [a] input.



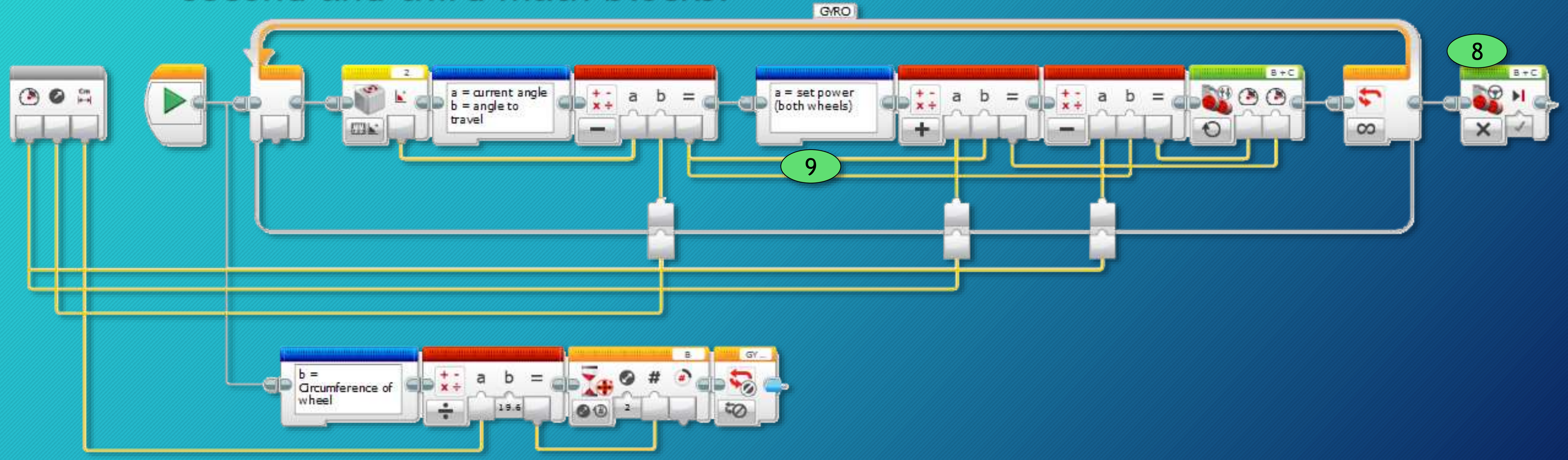
Vector Navigation

5. Insert a Math block set to add (+).
6. Insert a Math block set to subtract (-).
7. Insert a Tank Move block set to On. Set motor ports to B+C.



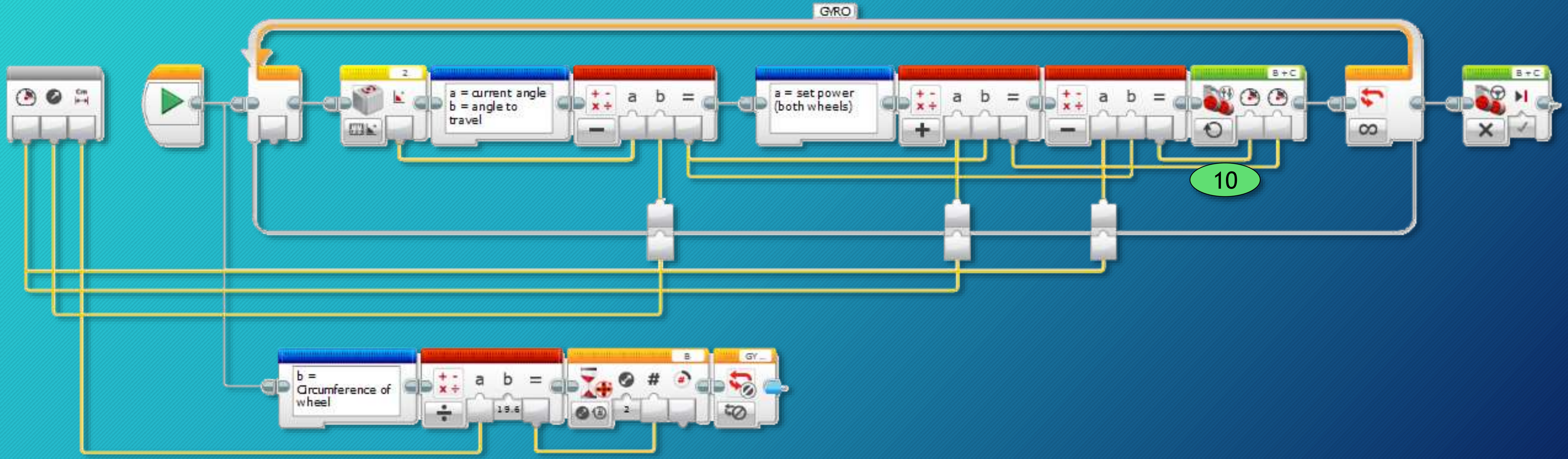
Vector Navigation

8. Insert a Move Steering block set to Off with motor ports B+C.
9. Drag two wires from the first Math block to the [b] inputs of the second and third math blocks.



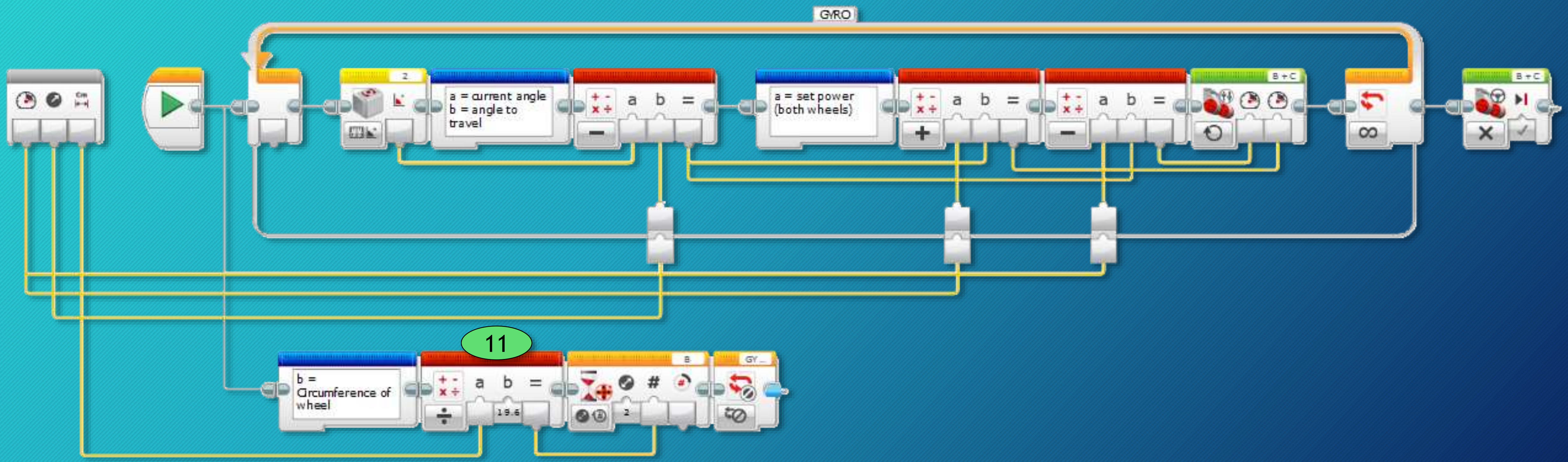
Vector Navigation

10. Drag a wire from the second Math block output to the C motor port of the Move Tank block. Drag a wire from the third Math block output to the B motor port of the Move Tank block.



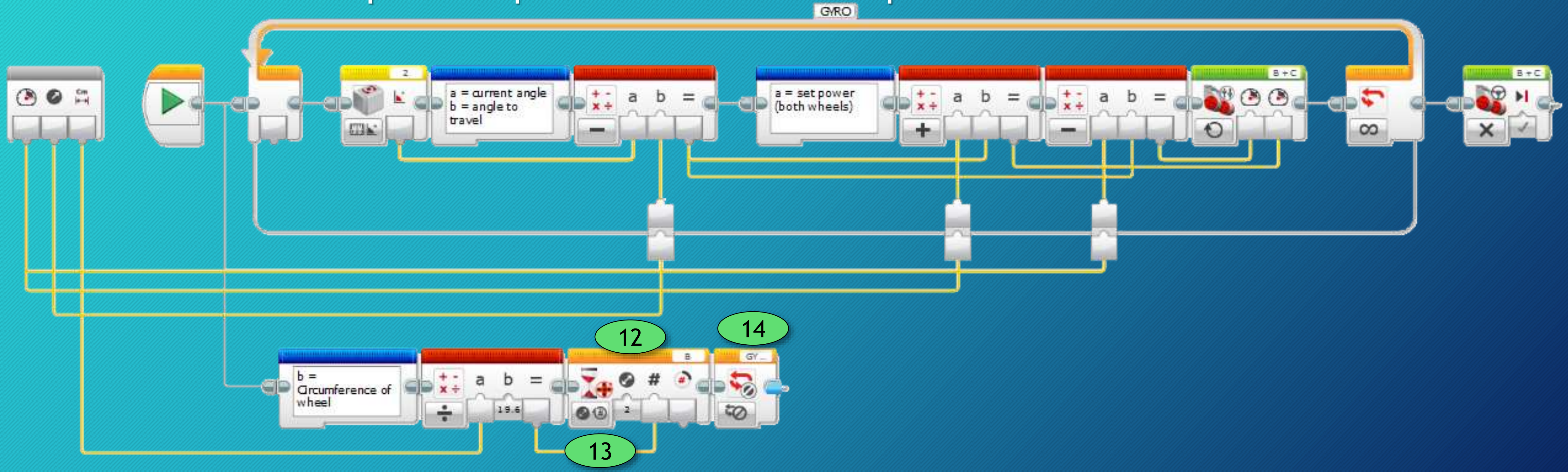
Vector Navigation

11. Insert a Math block on a separate track and set to divide (÷). Set [b] to the circumference of your robots wheels in centimeters.



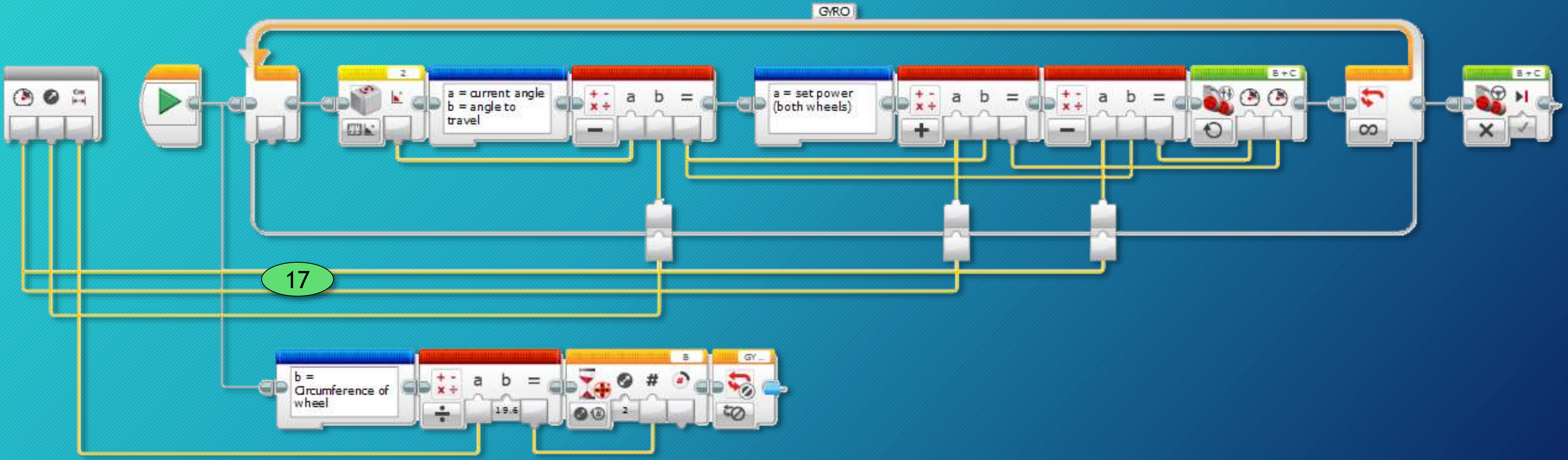
Vector Navigation

12. Insert a Wait block set to Motor Rotations ➔ Change ➔ Rotations.
13. Drag a wire from the Math block output [=] to the Wait block Amount [#].
14. Insert a Loop Interrupt block set to the loop above.



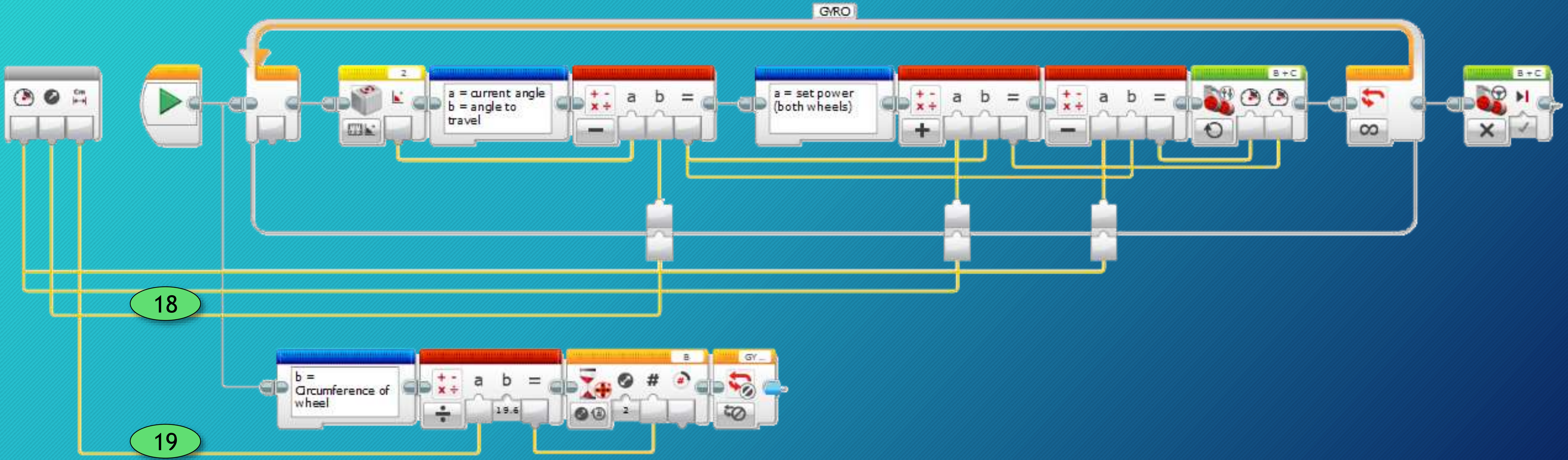
Vector Navigation

16. Create a My Block with three parameters, Power (Speed), Angle, and Distance.
17. Drag a wire from the Power parameter to the second Math block [a] input and a second wire from the Power parameter to the third Math block [a] input.



Vector Navigation

18. Drag a wire from the Angle parameter to the first Math block [b] input.
19. Drag a wire from the Distance parameter to the Math block [a] input in the separate track.



Vector Navigation Breakdown

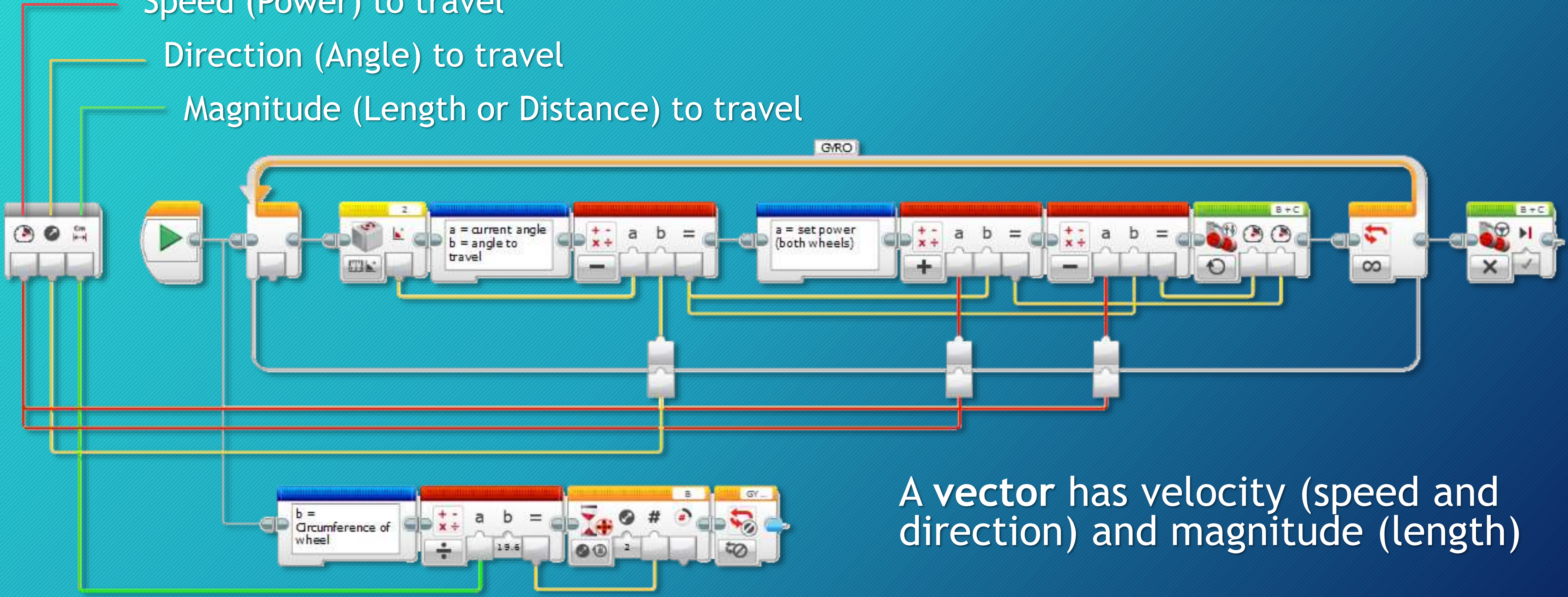
Understanding how Vector Navigation works

Vector Navigation (Speed, Direction, Magnitude) inputs

Speed (Power) to travel

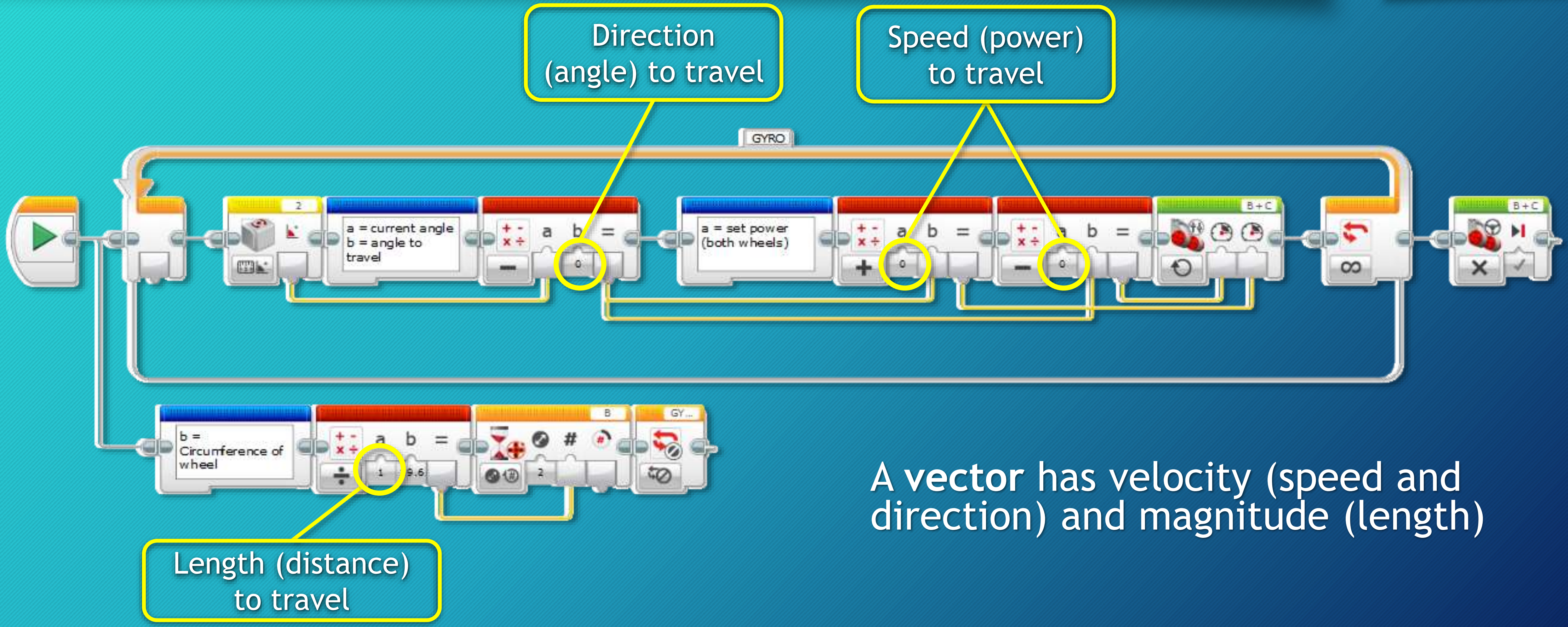
Direction (Angle) to travel

Magnitude (Length or Distance) to travel



A **vector** has velocity (speed and direction) and magnitude (length)

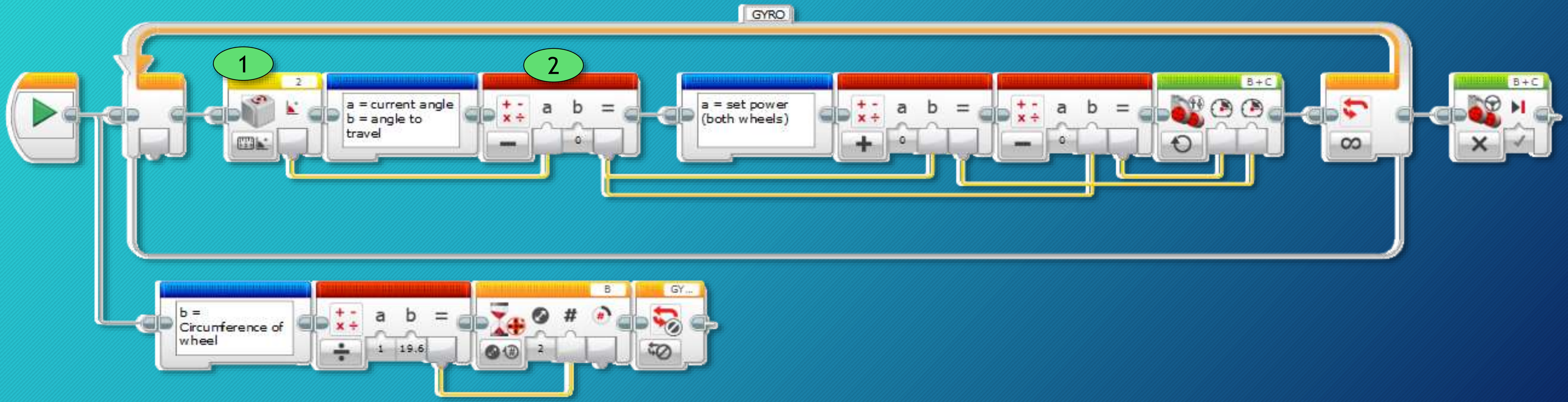
Vector Navigation Parameters (Inputs)



A **vector** has velocity (speed and direction) and magnitude (length)

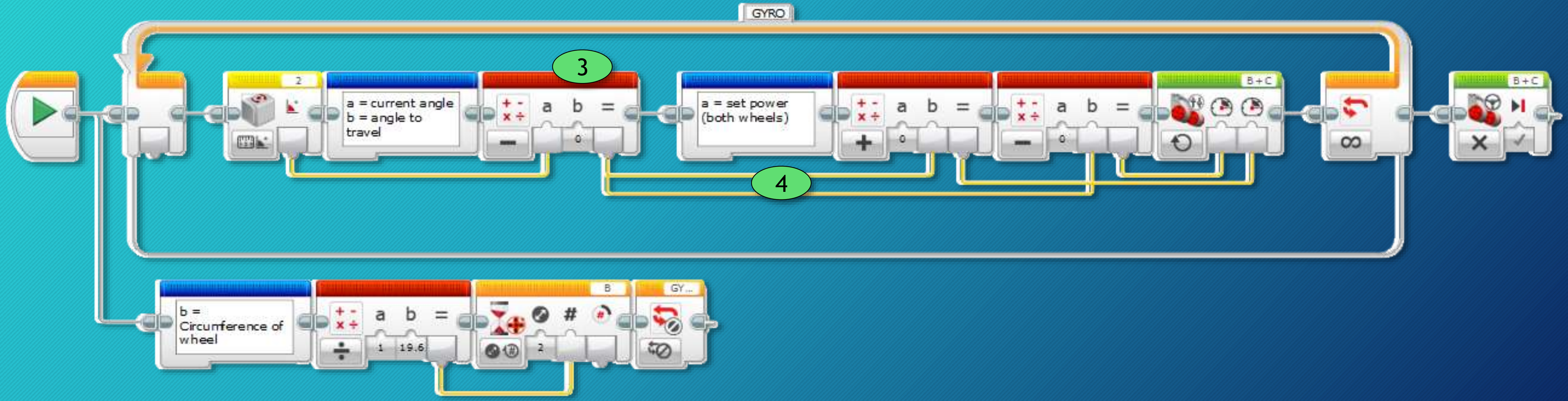
Vector Navigation Processing

1. Read the current Gyro value.
2. Outputs the value to the Math block [a] input.



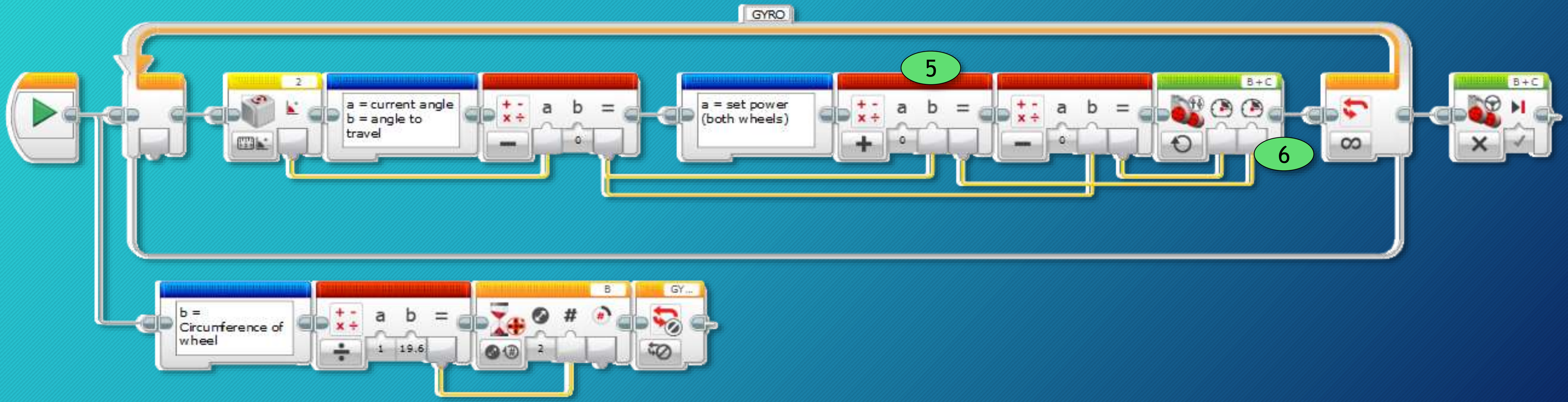
Vector Navigation Processing

3. First Math block subtract the current Gyro value from the direction to travel.
4. Outputs that value to both Math blocks [b] inputs.



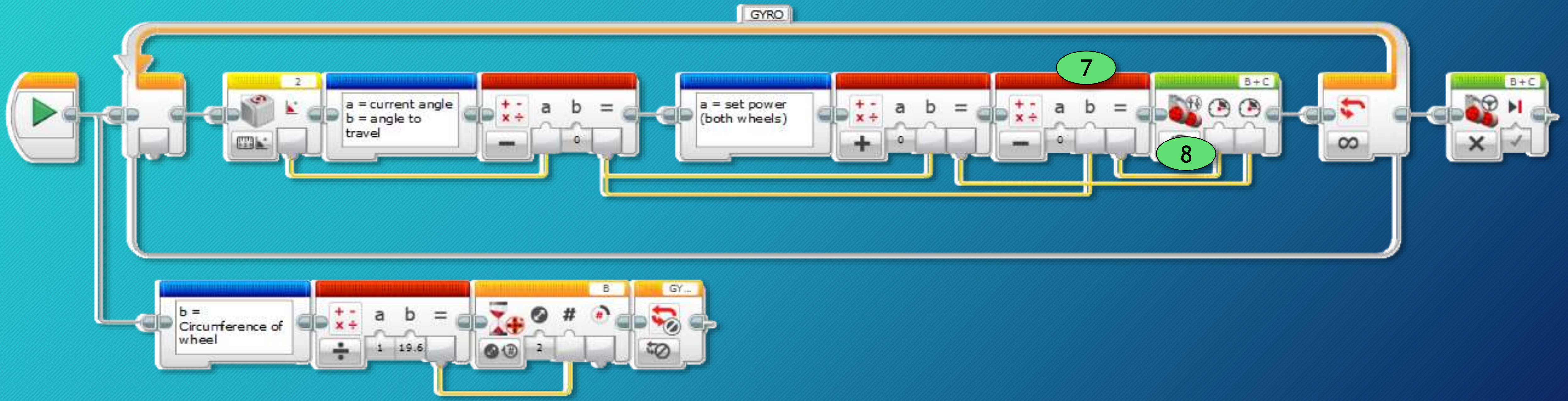
Vector Navigation Processing

5. Second Math block adds Gyro output to the power setting input [a].
6. Outputs the result to C input (speed) on the Move Steering block.



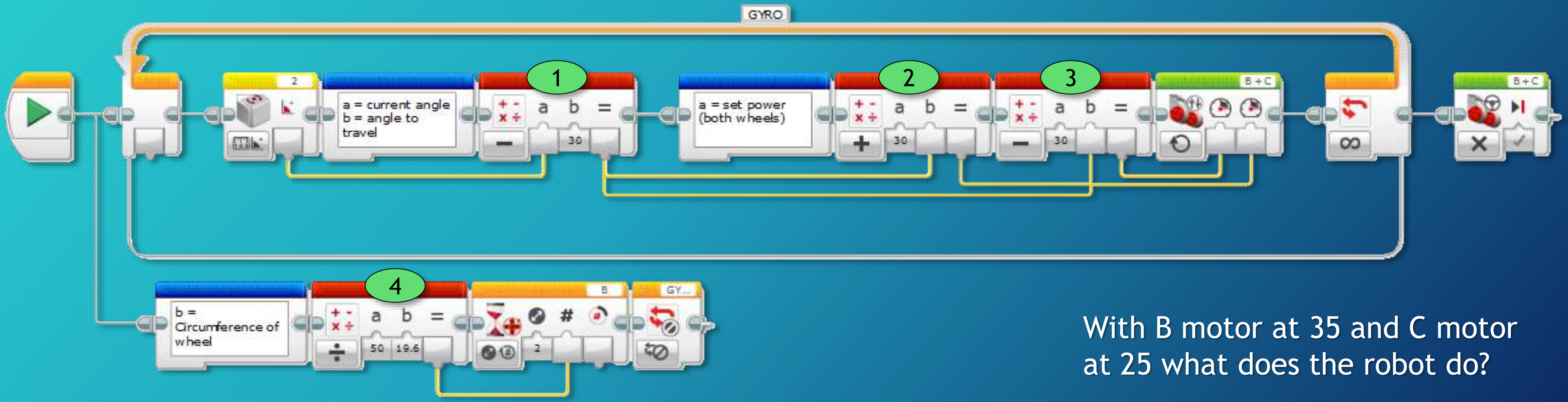
Vector Navigation Processing

7. Third Math block subtracts the Gyro output from the power setting [a].
8. Outputs the result to B input (speed) on the Move Steering block.



Vector Navigation – The Math

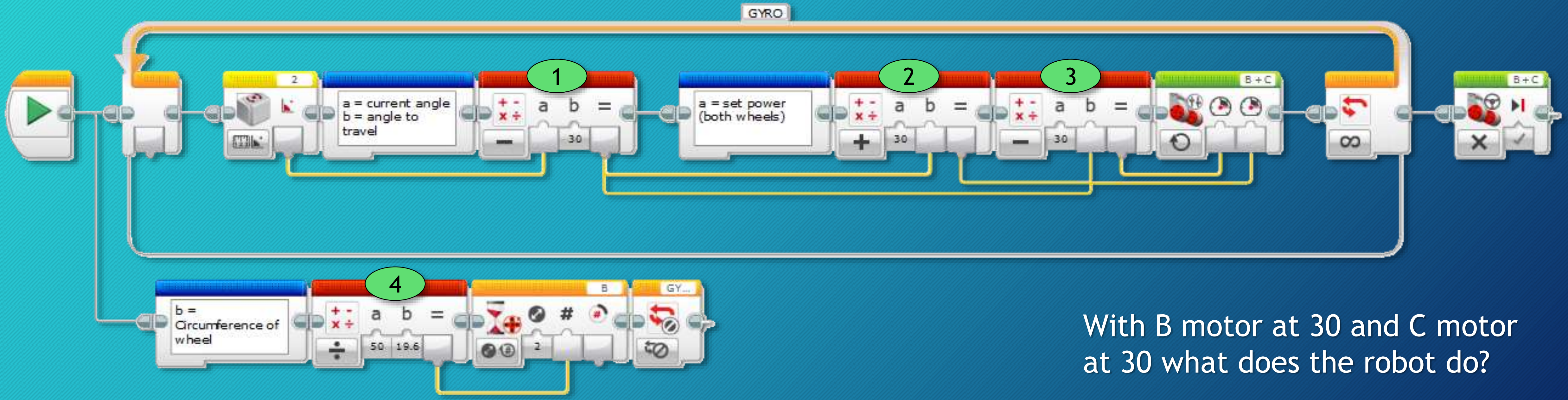
1. 35 (current gyro value) – 30 (angle to travel) = 5 (output)
2. 30 (set speed) + 5 input = 35 (output to C motor input)
3. 30 (set speed) – 5 input = 25 (output to B motor input)
4. 50 (set distance) ÷ 19.6 (wheel circumference) = 2.55 (rotations)



With B motor at 35 and C motor at 25 what does the robot do?

Vector Navigation – The Math

1. 30 (current gyro value) – 30 (angle to travel) = 0 (output)
2. 30 (set speed) + 0 input = 30 (output to C motor input)
3. 30 (set speed) – 0 input = 30 (output to B motor input)
4. 50 (set distance) ÷ 19.6 (wheel circumference) = 2.55 (rotations)



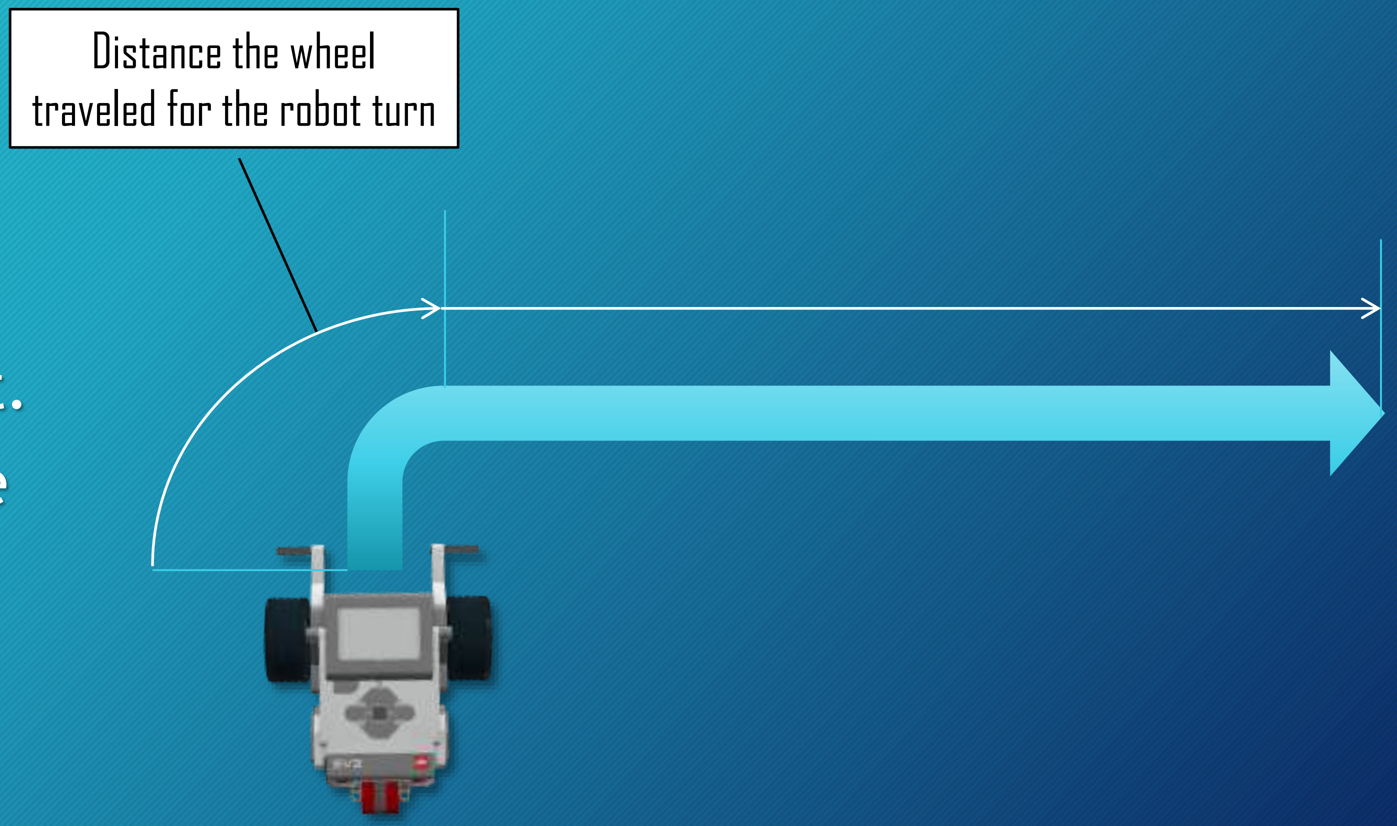
With B motor at 30 and C motor at 30 what does the robot do?

Vector Navigation - Issue

Issue: The length (distance) traveled includes the distance the measured wheel rotates during the turn.

Result: The distance the robot travels is less than the set distant.

Correction: Increase the distance setting. Consistency is the key, not the exact measurement.



Vector Navigation - Issue

Issue: The robot does not complete the full turn as set.

Result: When the robot needs to make a turn and travel a short distance the robot does not complete the turn because the distance is used on the turn.

Correction: In these circumstances use a left or right pin turn followed by a vector navigation My Block.

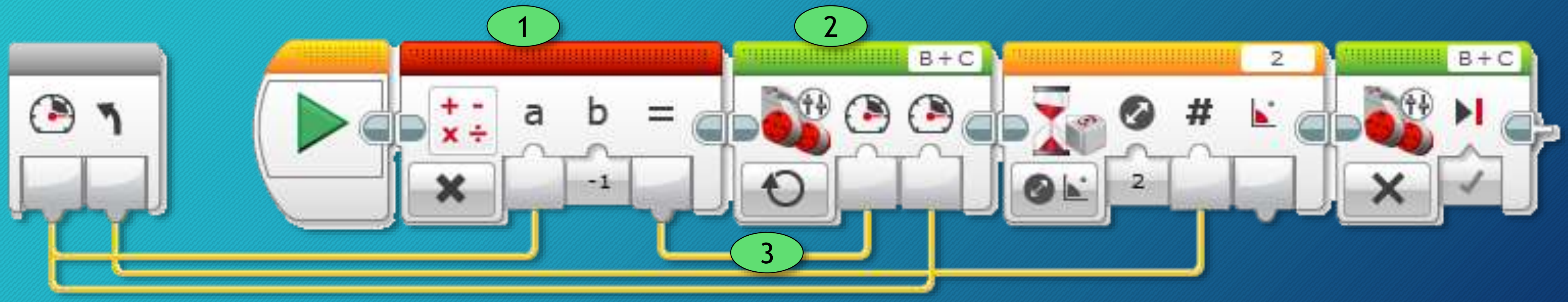
If the robot over or under rotates during the pin turn, the vector navigation My Block that follows will correct it.

Pin Turns

My Block programming instructions

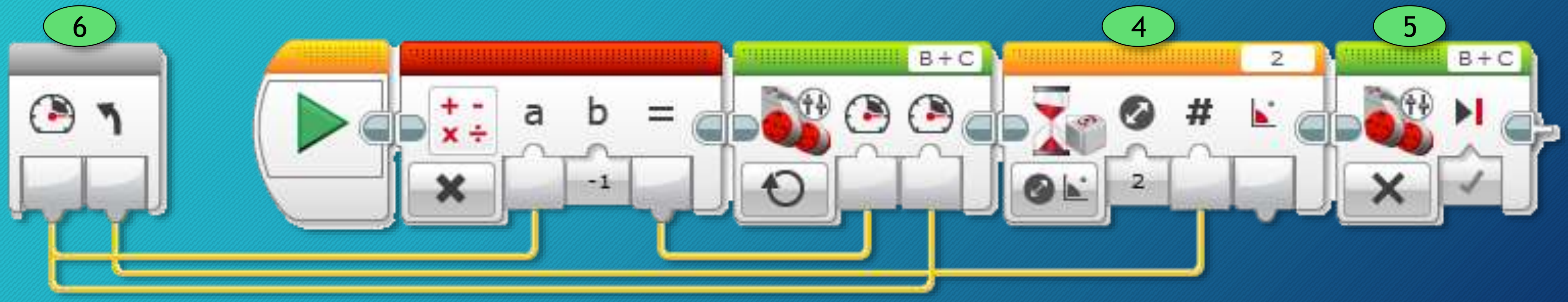
Pin Turn - Left

1. Insert Math block after the Start block, set Math block to multiply and “b” to -1.
2. Insert a Move Tank block set to On and port B+C.
3. Drag wire from Math block output (=) to Move Tank block B Power input.



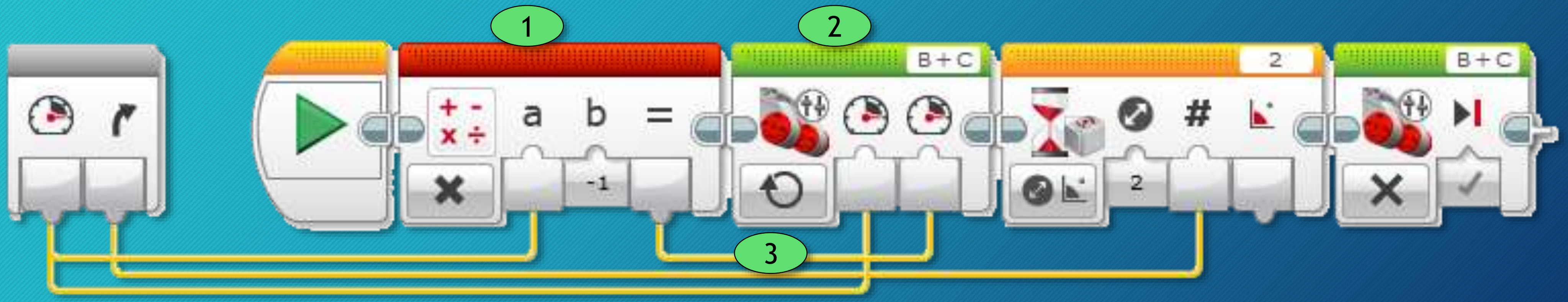
Pin Turn - Left

4. Insert a Wait block set to Gyro Sensor → Change → Angle, port 2.
5. Insert a Move Tank block set to Off and port B+C.
6. Convert to My Block with Power and Angle parameters.



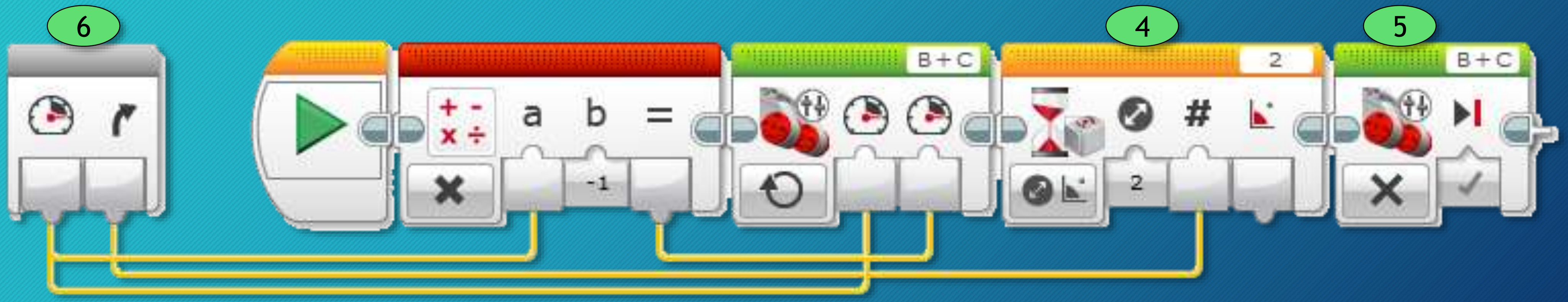
Pin Turn - Right

1. Insert Math block after the Start block, set Math block to multiply and b to -1.
2. Insert a Move Tank block set to On and ports B+C ports.
3. Drag wire from Math block output (=) to Move Tank block C input.



Pin Turn - Right

4. Insert a Wait block set to Gyro Sensor → Change → Angle.
5. Insert a Move Tank block set to Off and port B+C.
6. Convert to My Block with Power and Angle parameters.

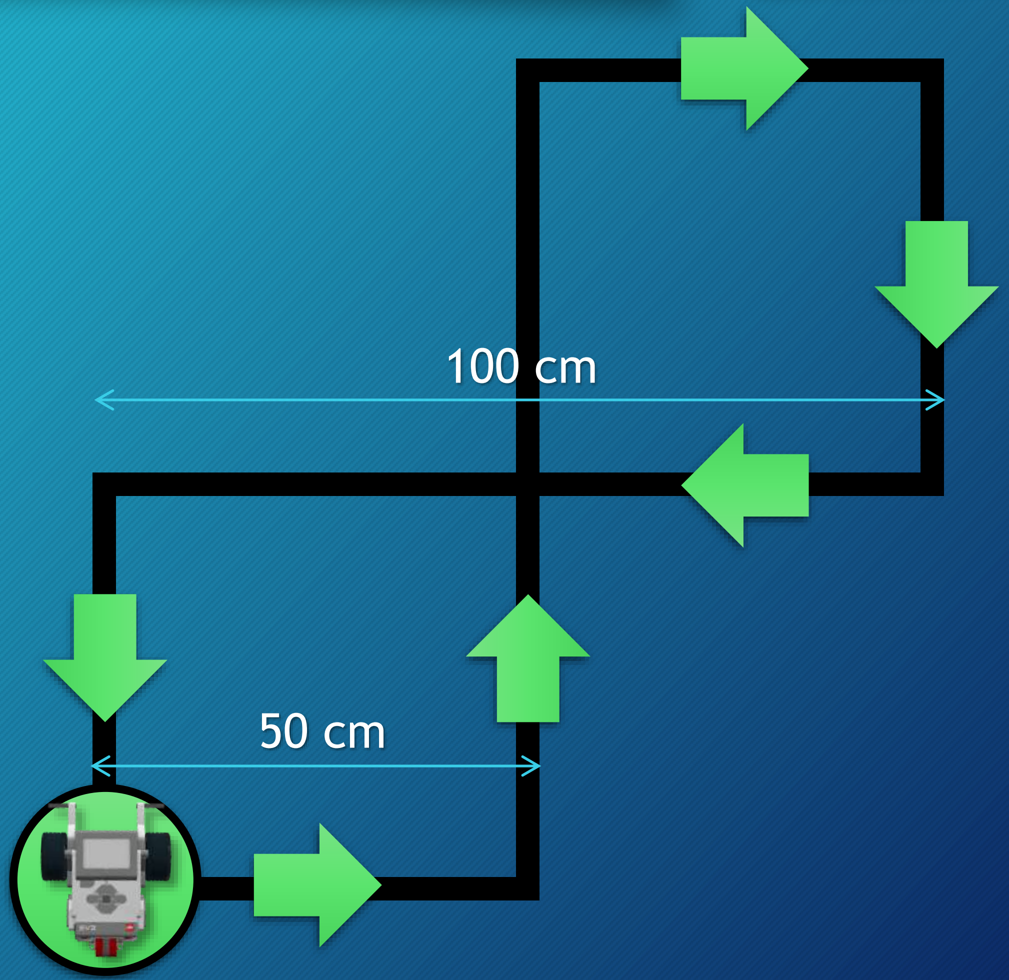


Using Vector Navigation

Programming example

Vector Navigation Challenge

Program a 100 cm square Figure 8 starting at the circle with the robot facing the direction shown.



Vector Navigation Example

Everything is awesome!

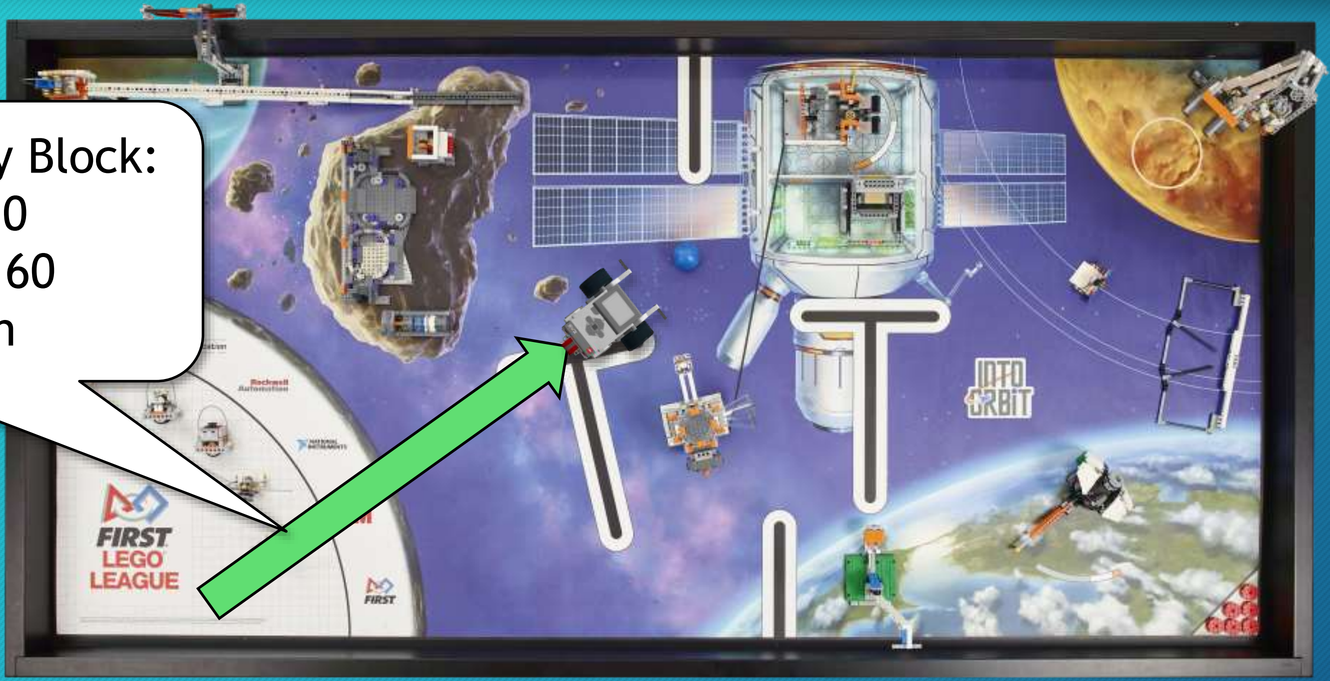
-Emmet Brickowski

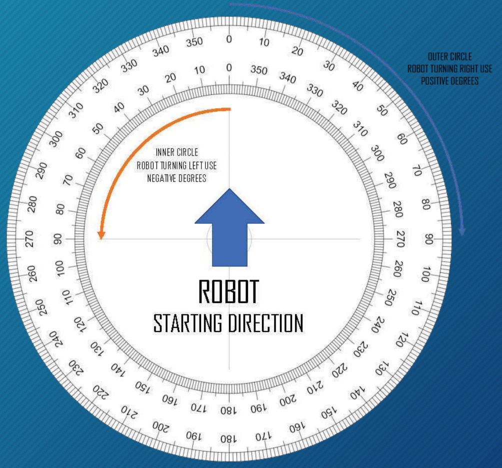
Email comments or correction to james.brodnick@gmail.com

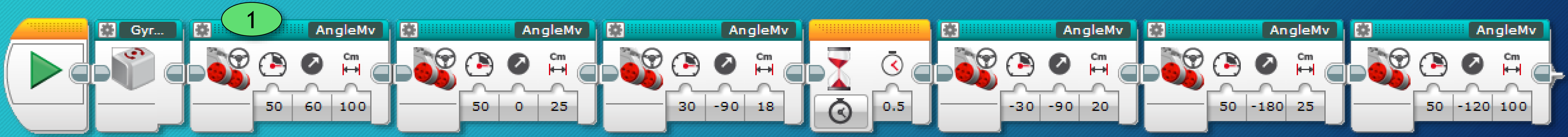
Vector Navigation Example

Vector Navigation Example

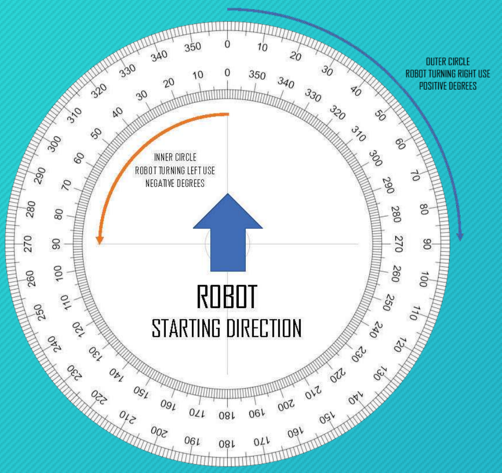
1. Vector Navigation My Block:
Speed (Power): 50
Direction (Angle): 60
Distance: 100 cm

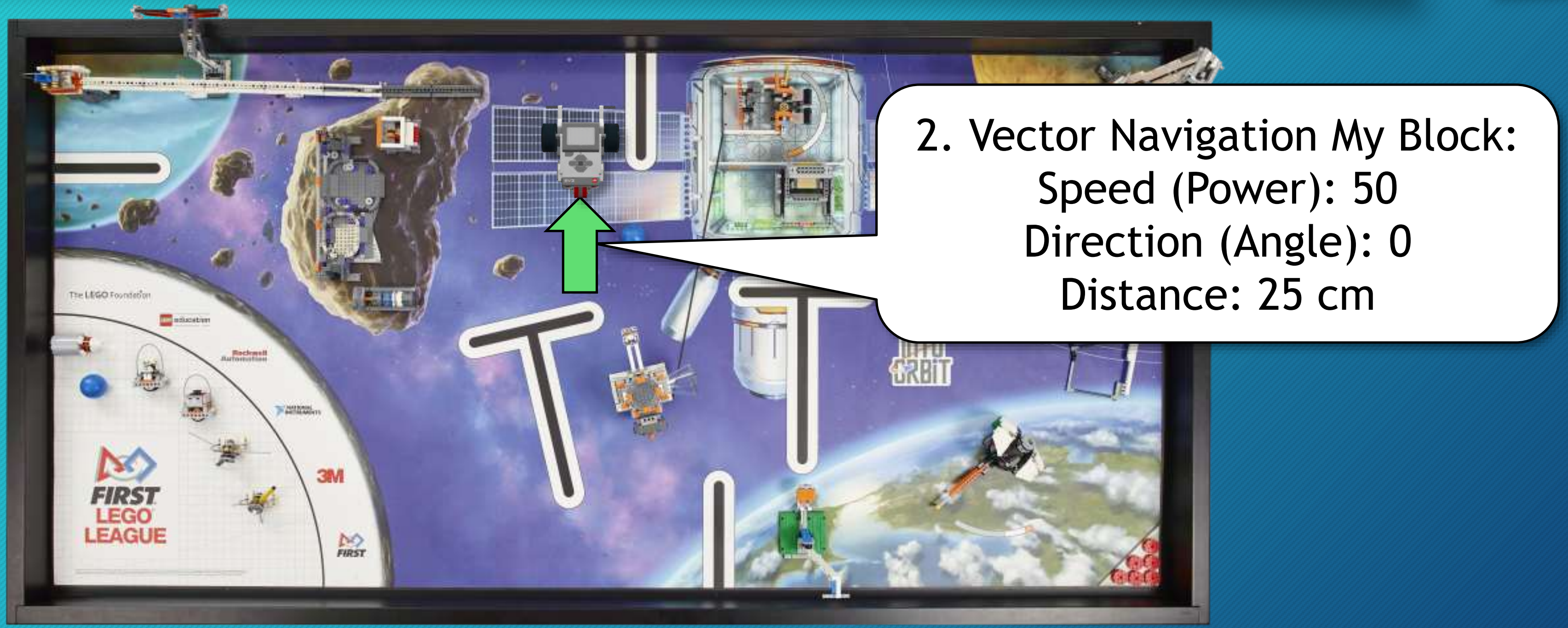


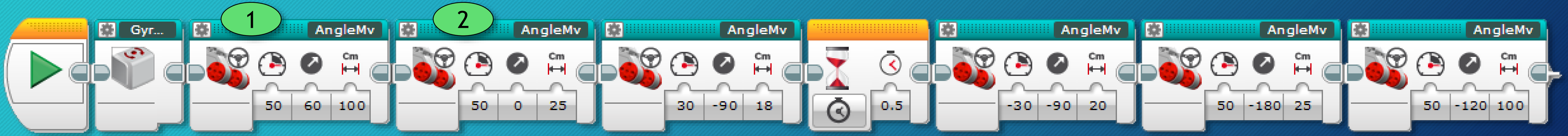




Vector Navigation Example

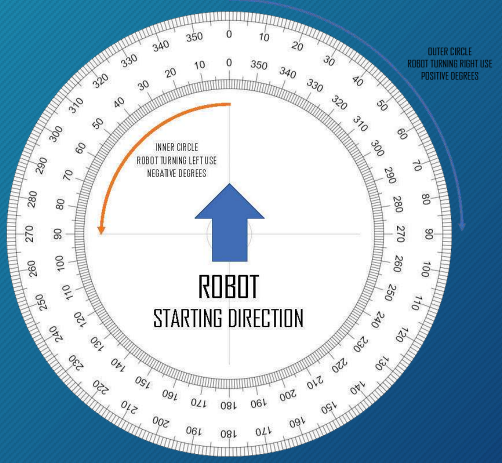


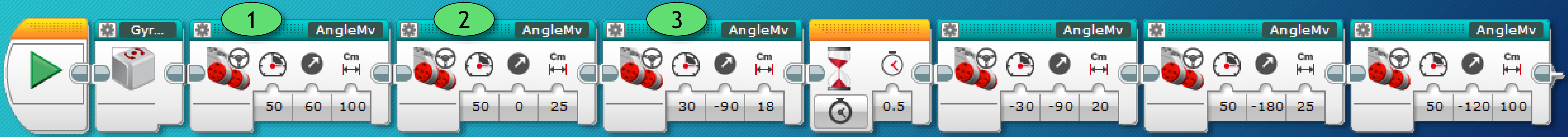




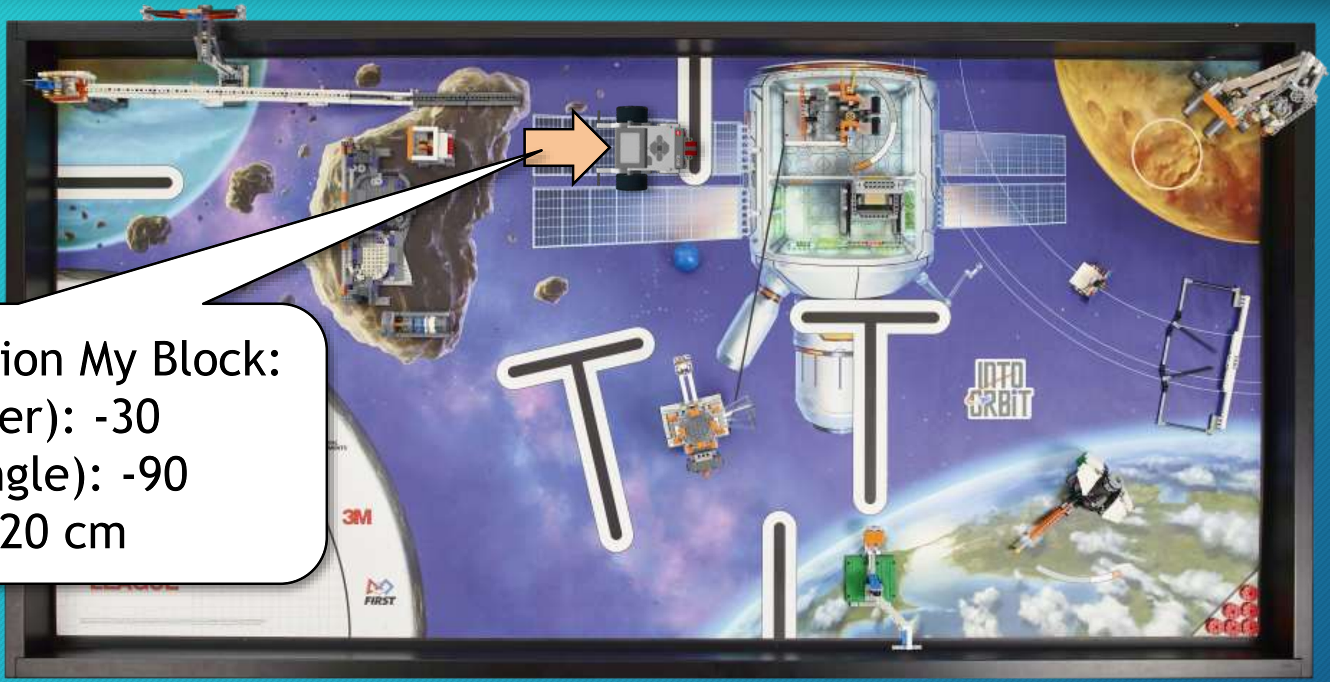
Vector Navigation Example

3. Vector Navigation My Block:
Speed (Power): 30
Direction (Angle): -90
Distance: 18 cm

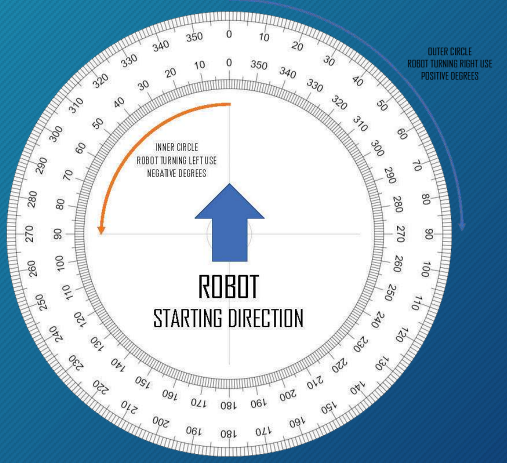


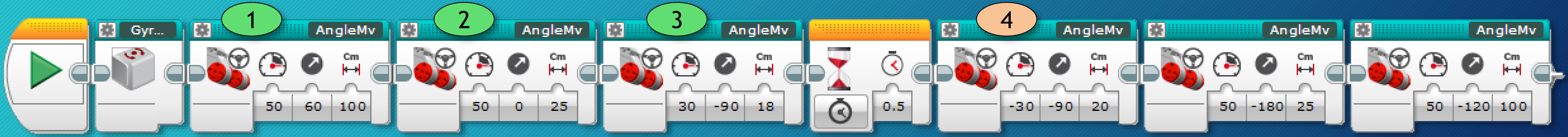


Vector Navigation Example

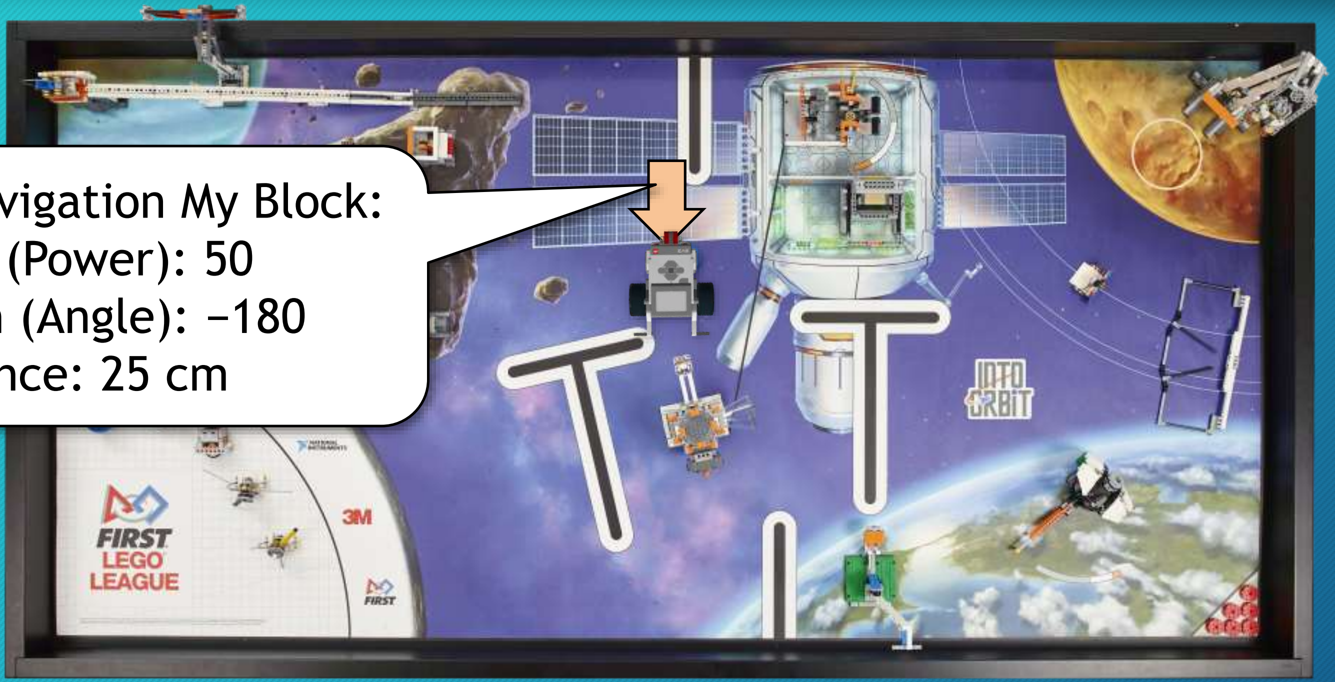


4. Vector Navigation My Block:
Speed (Power): -30
Direction (Angle): -90
Distance: 20 cm

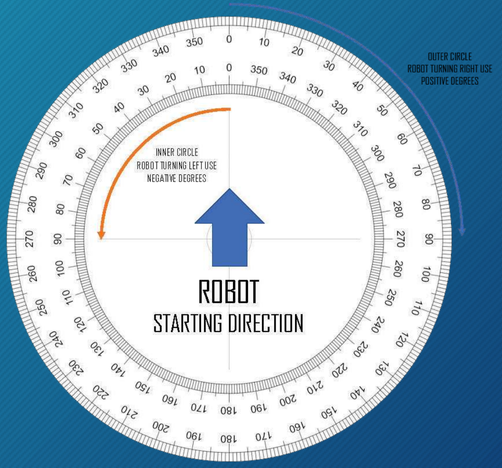


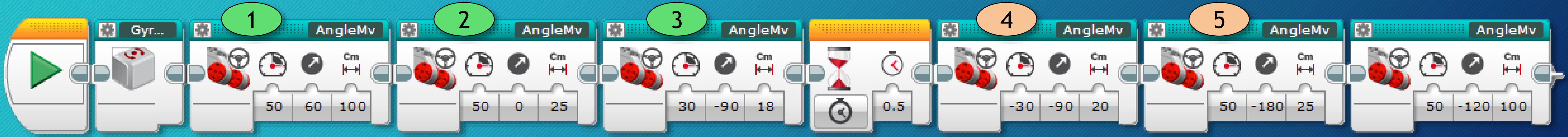


Vector Navigation Example



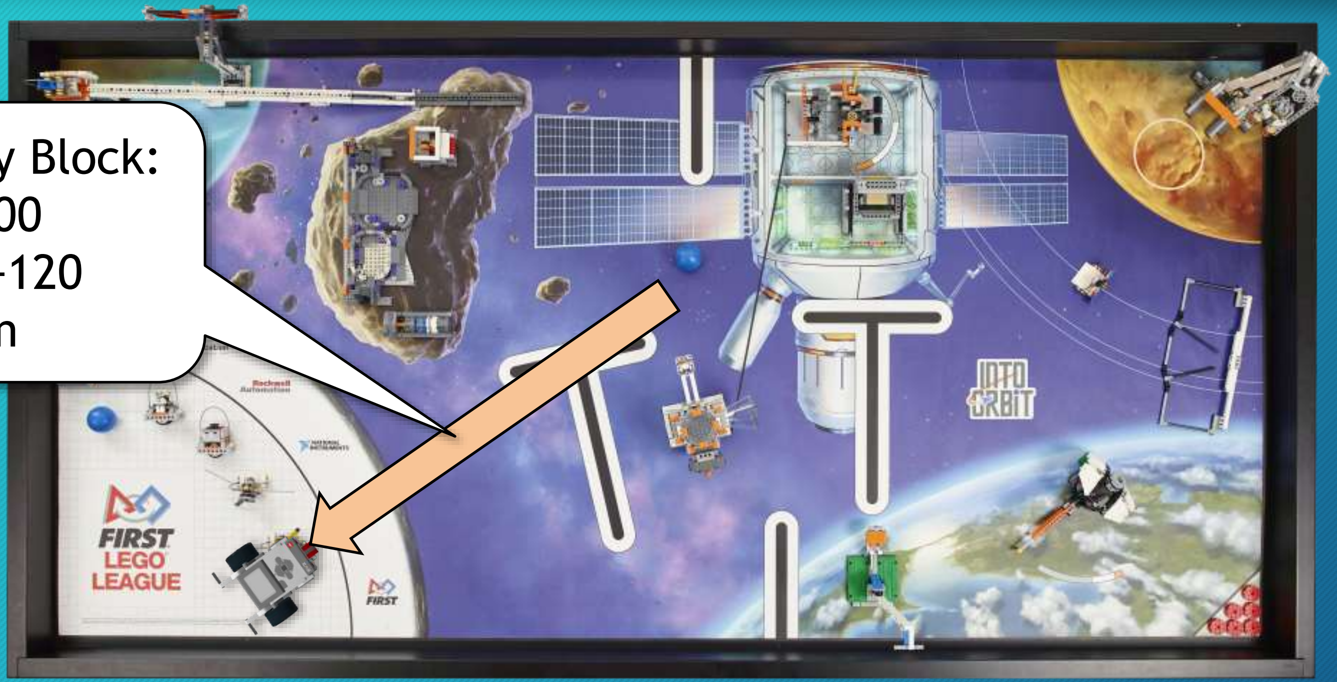
5. Vector Navigation My Block:
Speed (Power): 50
Direction (Angle): −180
Distance: 25 cm

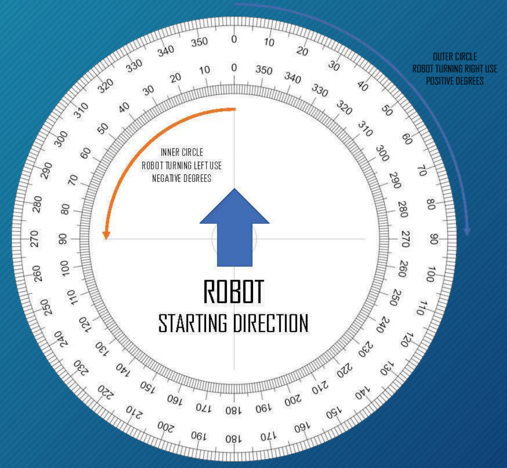


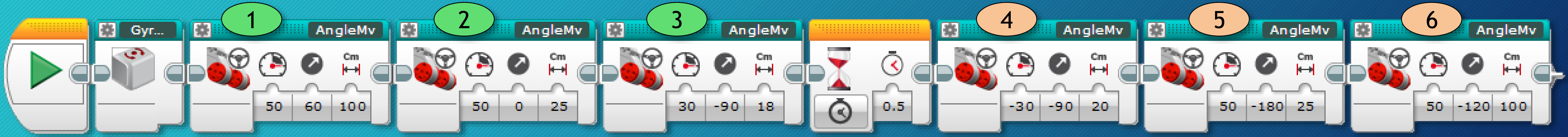


Vector Navigation Example

6. Vector Navigation My Block:
Speed (Power): 100
Direction (Angle): −120
Distance: 100 cm

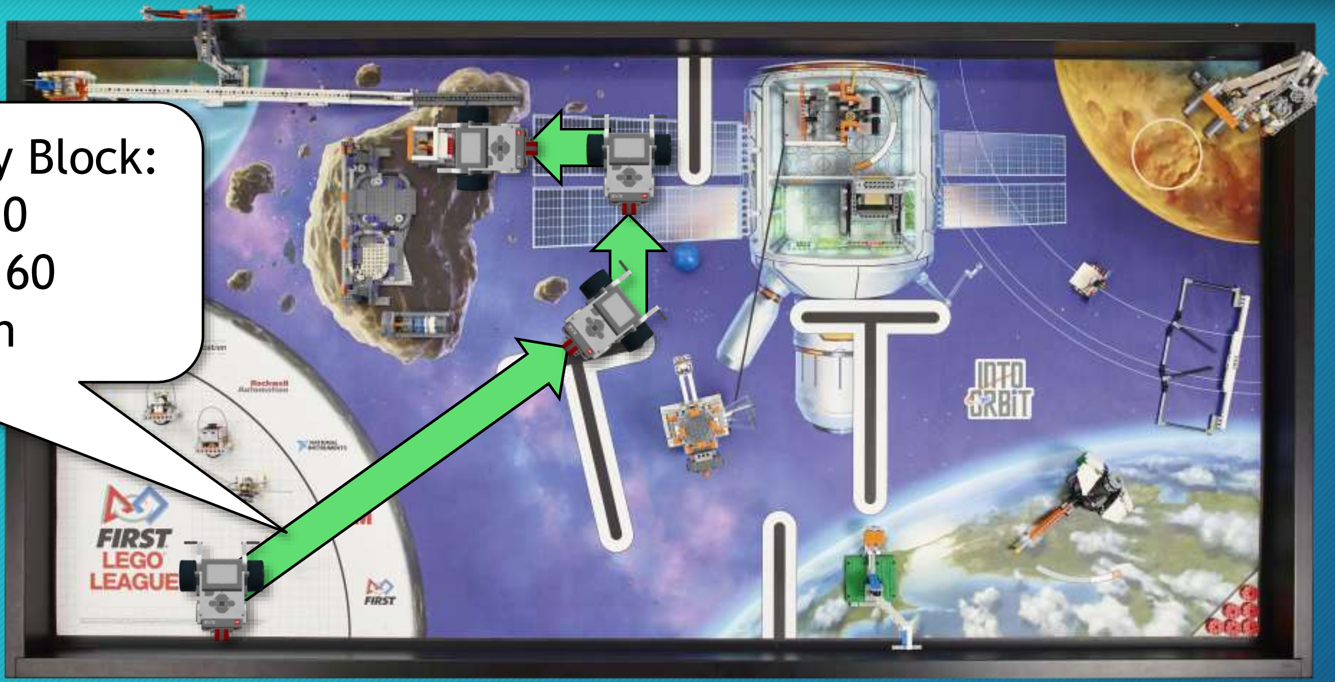


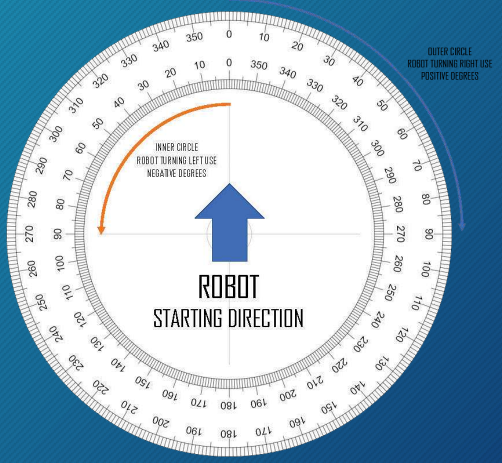


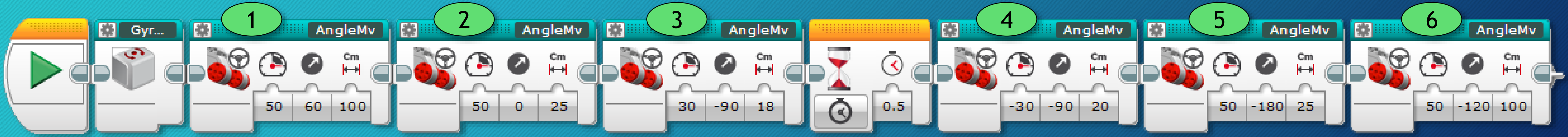


Vector Navigation Example

1. Vector Navigation My Block:
Speed (Power): 50
Direction (Angle): 60
Distance: 100 cm







Vector Navigation Example

1. Vector Navigation My Block:
Speed (Power): 50
Direction (Angle): 60
Distance: 100 cm