



Moving the RVR

Robotics Curriculum
IT Adventures



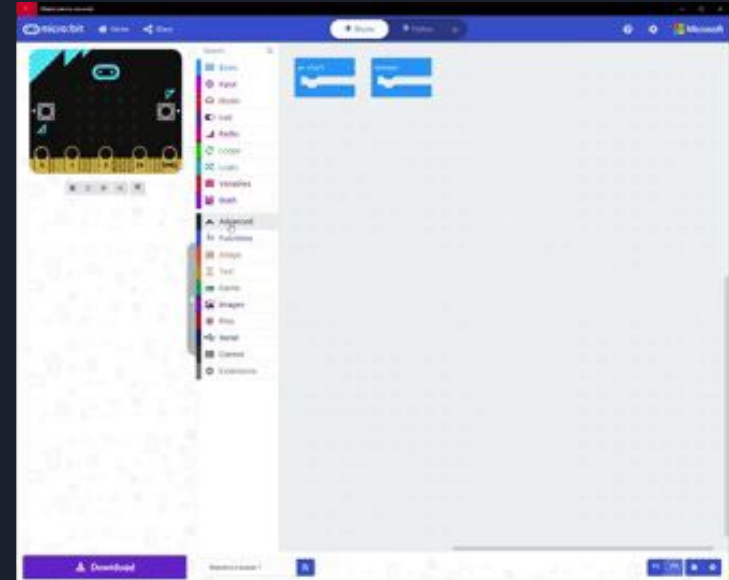
Lesson Overview

- Learn how to make a simple MakeCode program
- Learn how to interact with the RVR using the micro:bit
- Use our new knowledge to make the RVR move in certain ways!

Makecode Project

To make a new program for the micro:bit and RVR, follow these steps:

1. Open MakeCode and select New Project
2. Name the project according to the lesson, in this case “Robotics Lesson 1”, and click create
3. Import the RVR controls (Software Development Kit) by clicking Advanced and then Extensions
4. Make sure your computer is connected to the internet and search for RVR in the search bar.
5. Click on the result “sphero-sdk-microbit-mak...”
6. Start coding!

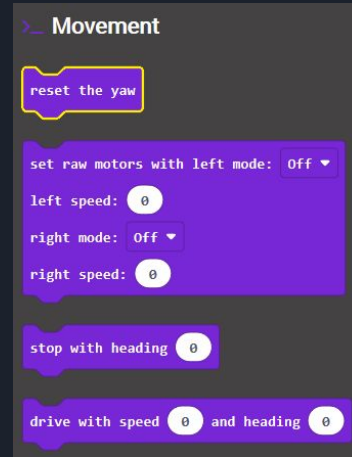


Sphero SDK Overview

- The Sphero RVR has three main sections for programming:
 - Movement is used to drive the RVR
 - Lights is used to control all the lights on the robot
 - Power is used to put the RVR in a sleep or active state



- We'll focus on Movement for now:
 - reset the yaw
 - set raw motors
 - stop with heading
 - drive with speed and heading



Movement Examples: Preserving Direction

Implement the following code and run it twice by pressing the reset button after the initial run, what happens?

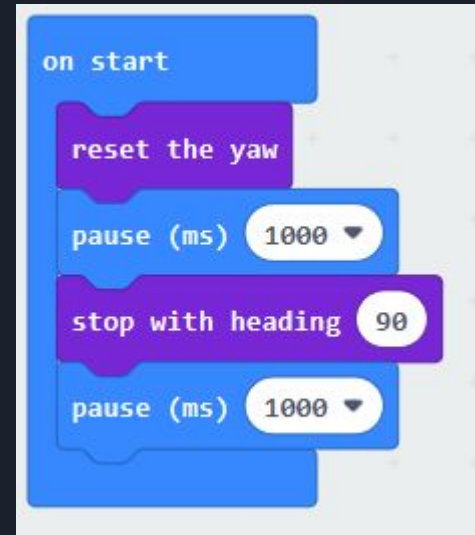


Try running the program a couple more times, but change the direction the RVR is looking each time, what happens?

Movement Examples: Reset the Yaw

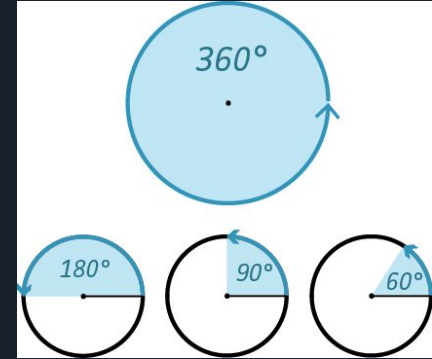
You should have noticed the RVR only turned once and always turned to a certain direction, even though you tried running the program more than once and with the RVR facing different directions. This is because the RVR remembers its initial direction and will move in respect to that.

If we use the reset yaw block, we will reset the initial direction to what the RVR is currently facing. Implement the code below and note any changes when running the program multiple times.



Movement Examples: Stopping and Moving

In the previous example, we used stop with heading to turn the RVR. Heading can be thought of as a circle. This is how we'll control which direction we want to drive in. We can input a value from 0 to 359.



Try using the drive with speed block now. The speed is controlled by putting in a value from -255 to 255. Make sure to test this program on the ground, and start off with a speed of 50. Then try some different speeds, including negative values.

For now, we'll skip the last movement block, set raw motors, but feel free to investigate what it does and learn how to make curved movements with it.



Primary Learning Challenge

You may have noticed that the RVR stops moving on its own after around two seconds. In order for it to move longer, we need to send another movement command to the robot. This is as simple as adding another block! Try and make the RVR move from one side of the room to the other, how many movement blocks does it take? Make sure to pause after every movement.

Now, mark where the RVR is on the floor, and make it move in a square using the what you've learned about heading.



Secondary Learning Challenge

Lets ramp up the difficulty a bit. Can you figure out how to make the RVR move in the pattern below?

