Programming the EV3 with Swift Playgrounds Lesson 2 - Straight Move

By Ahmad Sahar

Hi, It's Shah again. For those who don't know, I'm a professional trainer specialising in MacOS and iOS software and hardware, and I also conduct classes on iOS App Development and LEGO® Mindstorms.

In the last lesson, we wrote a simple program to display text on the screen. In this lesson we're going to build a robot and make it move forwards and backwards.

What you need:
- An iPad with Swift Playgrounds installed
- LEGO® Mindstorms EV3 Education (set no. 45544) or Home (set no. 31313)

Before you begin

For this lesson you need to build either the Educator robot from the 45544 set or DrivingBas3 from the 31313 set.

The instructions for the Educator robot can be found here: http://robotsquare.com/wp-content/uploads/2013/10/45544_educator.pdf

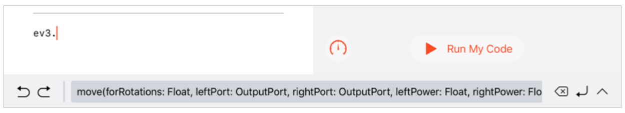
The instructions for DrivingBas3 can be found here: http://ev3lessons.com/RobotDesigns/instructions/DrivingBas3.pdf

Make sure you have fresh batteries in your robot, your iPad is fully charged, and your iPad is paired to your robot.

Straight Move

You can use the template you used in the last lesson, or you can download a fresh copy. Remove all the code from the template, and connect to the EV3 brick.

Tap ev3 from the suggestions list. It appears on the page. Tap the dot in the suggestions list. Scroll through all the suggestions in the suggestions list until you see this one that begins with "move":



```
ev3.
```

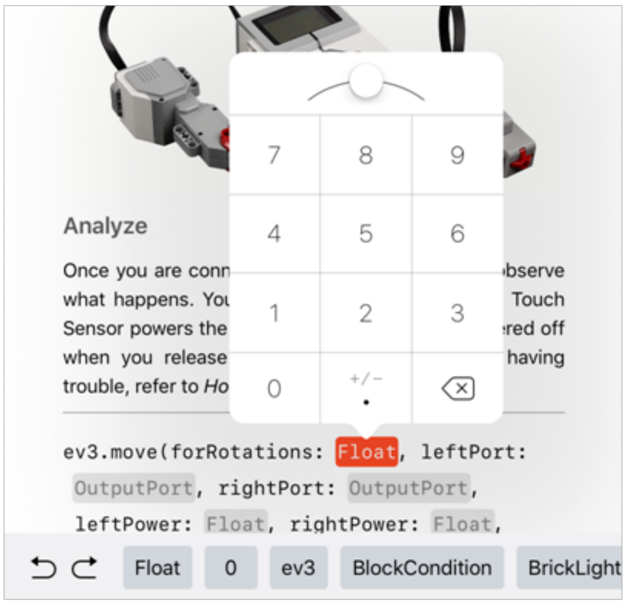
```
move(forRotations: Float, leftPort: OutputPort, rightPort: OutputPort, leftPower: Float, rightPower: Flo
```

Tap it to insert into the page.

This is a Swift function which is similar to the EV3-G Move Tank block.
It has a number of parameters:
forRotations: Number of rotations to execute
leftPort: The port the left motor is connected to. Can be .a, .b, .c or .d
rightPort: The port the right motor is connected to. Can be .a, .b, .c or .d
leftPower: The power level specified for leftPort.
rightPower: The power level specified for rightPort.
brakeAtEnd: Brake or coast at the end of the move.

In general, to change the value for the parameters, tap a parameter, and you can either use a picker to enter the value, or choose from the suggestions list.

We'll start by changing the forRotations parameter. Tap the parameter, which is represented by the word Float. When the picker appears, choose 1. This sets the number of rotations to be executed to 1.



Analyze

Once you are conn... bserve what happens. You... Touch Sensor powers the... red off when you release... having trouble, refer to Ho

```
ev3.move(forRotations: Float, leftPort:
 OutputPort, rightPort: OutputPort,
 leftPower: Float, rightPower: Float,
```

Tap the *leftPort* parameter, *OutputPort* and tap the dot on the suggestions list.

After you have done so, the motor ports a, b, c and d appear in the suggestions list. Choose b. This sets *leftPort* to Port B on the programmable brick.

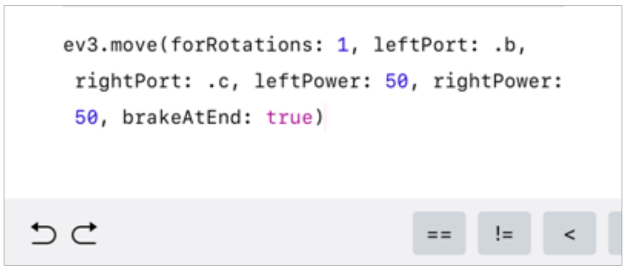
Tap the *rightPort* parameter, *OutputPort* and tap the dot on the suggestions list. Choose c from the suggestions list. This sets *rightPort* to Port C on the brick.

Tap the *leftPower* parameter, *Float*. The picker appears. Enter 50. This sets the power level of the left motor to 50.

Tap the *rightPower* parameter, *Float*. The picker appears. Enter 50. This sets the power level of the right motor to 50.

Finally, tap the *brakeAtEnd* parameter, *Bool*. Tap true from the suggestions list.

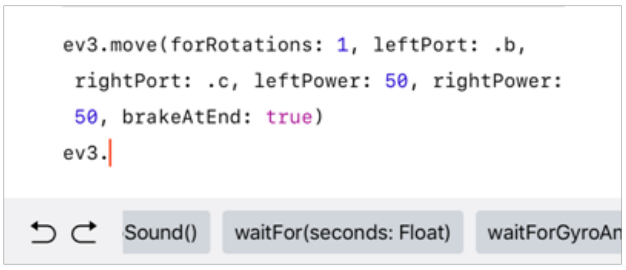
The function should look like this at this point:



```
ev3.move(forRotations: 1, leftPort: .b,
 rightPort: .c, leftPower: 50, rightPower:
 50, brakeAtEnd: true)
```

Next we're going to make the robot pause using a wait function.

Tap on the space below the code you've entered and choose ev3 from the suggestions list. Tap the dot, and scroll through the list of suggestions until you find *waitFor(seconds: Float)*. Tap it.



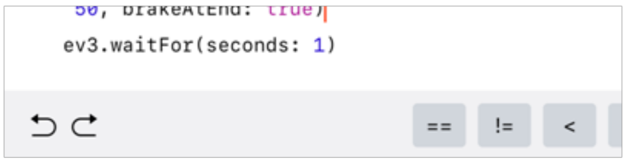
```
ev3.move(forRotations: 1, leftPort: .b,
 rightPort: .c, leftPower: 50, rightPower:
 50, brakeAtEnd: true)
ev3.
```

This is a Swift function which is similar to the EV3-G Wait block.

It has one parameter, seconds, which is the number of seconds to wait.

Tap the *seconds* parameter, *Float*. Choose 1 from the picker to make the program pause for 1 second.

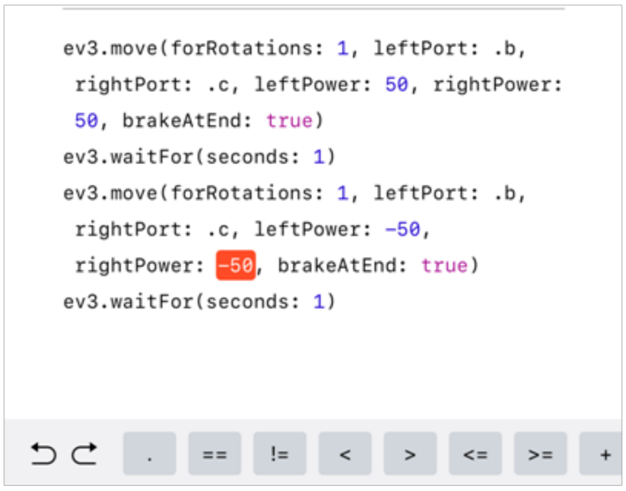
The function should look like this:



```
 50, brakeAtEnd: true)
ev3.waitFor(seconds: 1)
```

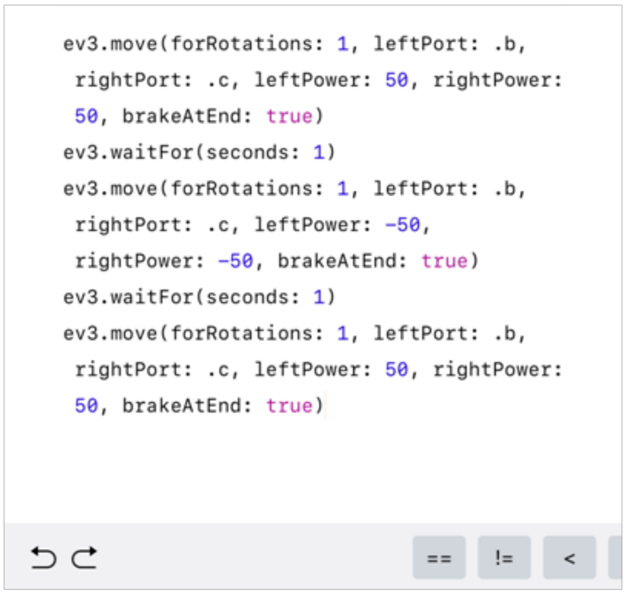
Select all the code you've typed in so far, and copy it. Tap below the existing code and paste.

To make the robot go backwards, we're going to set the power values of the second move function to -50 for both motors. Tap the *leftPower* parameter for the second move function, and when the picker appears, look for the key that has +/- on it. Tap and drag downwards, and you'll see the value has been changed to -50. Do the same thing for the *rightPower* parameter. The result should be as follows:



```
ev3.move(forRotations: 1, leftPort: .b,
 rightPort: .c, leftPower: 50, rightPower:
 50, brakeAtEnd: true)
ev3.waitFor(seconds: 1)
ev3.move(forRotations: 1, leftPort: .b,
 rightPort: .c, leftPower: -50,
 rightPower: -50, brakeAtEnd: true)
ev3.waitFor(seconds: 1)
```

Tap below the code you've typed in so far and paste again. Remove the last line of code, and you should get this:



```
ev3.move(forRotations: 1, leftPort: .b,
 rightPort: .c, leftPower: 50, rightPower:
 50, brakeAtEnd: true)
ev3.waitFor(seconds: 1)
ev3.move(forRotations: 1, leftPort: .b,
 rightPort: .c, leftPower: -50,
 rightPower: -50, brakeAtEnd: true)
ev3.waitFor(seconds: 1)
ev3.move(forRotations: 1, leftPort: .b,
 rightPort: .c, leftPower: 50, rightPower:
 50, brakeAtEnd: true)
```

Run the program. The robot should move forward by one rotation, pause for one second, move backward for one rotation, pause for one second and move forward by one rotation again.

Great job! We've come to the end of the lesson. In the next lesson we'll make the robot turn.

If you wish to know more about me and what I do, feel free to visit my company website, http://tomafuwi.tumblr.com, like my Facebook page at http://facebook.com/tomafuwi, or follow me on Twitter at https://twitter.com/shah_apple

All the best and take care.
#