

LESSON 5: STOP AT OBJECT

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 made our robot interact with an object. In this lesson, we're going to make our robot stop when it senses something in front of it.

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 the DrivingBas3 from the 31313 set.

The instructions for the Educator robot can be found in the Education version of the LEGO Mindstorms software, or can be downloaded here: robotsquare.com/wp-content/uploads/2013/10/45544_educator.pdf

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

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

You'll also need to attach a sensor to your robot. If you have the 45544 set, connect the ultrasonic sensor using the instructions in the PDF shown above. If you have the 31313 set, connect the infrared sensor to your robot. You can use the Educator robot PDF as a guide, but note that some of the parts shown are not included in the 31313 set; you'll have to substitute with other parts to attach the sensor.

Ultrasonic or Infrared Sensor

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.

If you have a 45544 set, enter the following code:

```
ev3.move(leftPort: .b, rightPort: .c,
leftPower: 50, rightPower: 50)
ev3.waitForUltrasonicCentimeters(on:
.four, lessThanOrEqualTo: 20)
ev3.stopMove(leftPort: .b, rightPort: .c,
withBrake: true)
```

If you have a 31313 set, enter the following code:

```
ev3.move(leftPort: .b, rightPort: .c,
leftPower: 50, rightPower: 50)
ev3.waitForIRProximity(on: .four,
lessThanOrEqualTo: 20)
ev3.stopMove(leftPort: .b, rightPort: .c,
withBrake: true)
```

The first line is a Swift function which is similar to the EV3-G Move Tank block. It is designed to work with the Driving Base robot. It has four parameters:

- **leftPort:** The port the motor driving the left wheel is connected to.
- **rightPort:** The port the motor driving the right wheel is connected to.
- **leftPower:** The power level specified for leftPort.
- **rightPower:** The power level specified for rightPort.

This line will start the robot moving forward with the power set to 50% for both left and right motors.

The second line is a Swift function which

is similar to the EV3-G Wait block. It has two parameters:

- **on:** The port the ultrasonic or infrared sensor is connected to. Usually this will be port four.
- **lessThanOrEqualTo:** The sensor will measure the distance between itself and anything in front of it. This line will execute when the threshold has been reached (20 in this case).

The third line is a Swift function which is similar to the EV3-G Move Tank Block. It has three parameters:

- **leftPort:** The port the motor driving the left wheel is connected to.
- **rightPort:** The port the motor driving the right wheel is connected to.
- **withBrake:** Stops the motors connected to leftPort: or rightPort: and brakes them instead of letting them coast.

Point the robot towards a wall. Run the program. The robot should move forward and stop just before it hits the wall.

Great job! We've come to the end of the lesson. In the next lesson, we'll program the touch sensor to stop the robot when pressed.

If you wish to know more about me and what I do, feel free to visit my company website, tomafuwi.tumblr.com, like my Facebook page at facebook.com/tomafuwi, follow me on Twitter at twitter.com/shah_apple and subscribe to my YouTube channel at youtube.com/c/CoderShah.

All the best and take care.

The screenshot shows the Swift Playgrounds app interface. On the left, there is a code editor with the following Swift code:

```
ev3.move(leftPort: .b, rightPort: .c,
leftPower: 50, rightPower: 50)
ev3.waitForIRProximity(on: .four,
lessThanOrEqualTo: 20)
ev3.stopMove(leftPort: .b, rightPort: .c,
withBrake: true)
```

Below the code, there is an "Analyze" section with the text: "Once you are connected, tap Run my Code and observe what happens. You should see that pressing the Touch Sensor powers the Large Motor on. It will be powered off when you release the Touch Sensor. If you're having trouble, refer to How do I connect? for help."

On the right, there is a sensor data panel with four sensors:

- LARGE MOTOR:** 160 Degrees
- LARGE MOTOR:** 157 Degrees
- MEDIUM MOTOR:** 0 Degrees
- INFRARED:** 11 Proximity
- TOUCH:** 0 Touched
- LIGHT:** 61 Reflect

At the bottom, there is a "Run My Code" button and a status bar with various system parameters like true, DBL_TRUE_MIN, FLT_TRUE_MIN, LDBL_TRUE_MIN, SM_TRUESHARED, and SecTrustEvali.