

OrbBasic Lesson 3

Sensors and Random: Student Guide

Overview

In this lesson, you're going to create a new program with OrbBasic that makes the Sphero change colors when it senses it is in the air. You'll be using variables and conditional statements again, and you'll also learn about Sphero's accelerometer and the accelone and random functions.

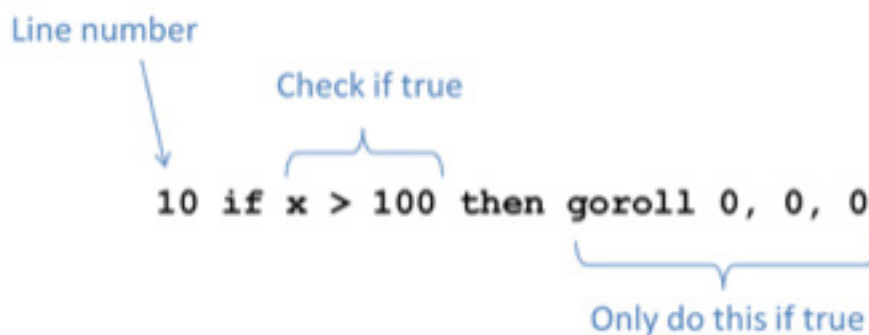
Here are the Sphero commands you'll be using for this lesson:

- goroll – Makes Sphero roll at a given speed and heading. Also makes it stop.
- delay – Makes Sphero wait an amount of time before doing the next command
- goto – Makes Sphero go to a certain place in the program
- variables – Used to store a number
- if/then – Used to make the Sphero do something if a statement is true
- if/then/else – Like if/then, but also does something else if the statement is false
- LEDC – Makes the Sphero light up a color
- Accelone - returns the effective acceleration vector that Sphero is experiencing.
- Heading- the direction Sphero will move, range 0-359 degrees
- Rnd- returns a random number between 1 and the value provided

First you have to connect Sphero to the iPad (Part 1), then you'll aim Sphero (Part 2), then you'll write an OrbBasic program to make Sphero turn purple when it is in the air (Part 3). Next, you'll make it light up with a different color when it collides with another object (Part 4). Finally, you'll have a challenge to use the accelone and random functions to make Sphero roll in a random direction after sensing a collision (Part 5).

A Quick Review

Let's go through a quick review of what you learned in the last lesson. The if/then statement looks like this:



A Quick Review (continued)

This line is saying that if the variable x is greater than 100, then Sphero should stop. (A goroll command with all zeros will stop the Sphero.) The part after if usually involves a variable. The part after then can be anything you want Sphero to do.

Also as a review from OrbBasic lesson 2, these are the colors that you can use with LEDC:

For example, a command to light the Sphero orange would be:

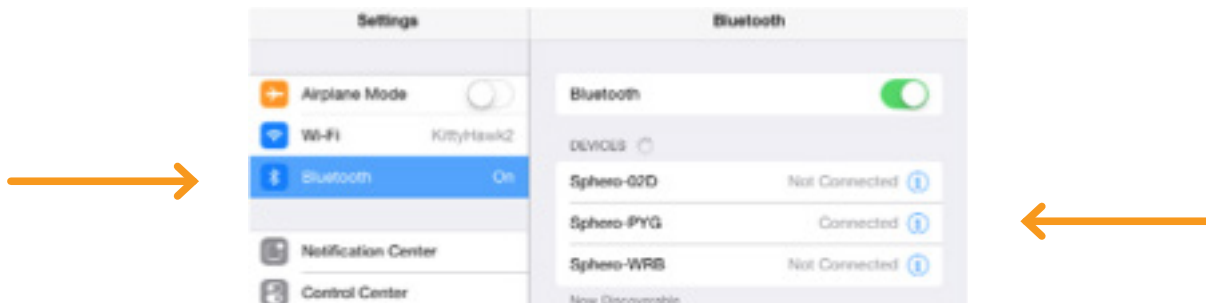
```
50 LEDC 4
```

Number	Color
0	No light
1	Red
2	Green
3	Blue
4	Orange
5	Purple
6	White
7	Yellow

Part 1: Connect the Sphero

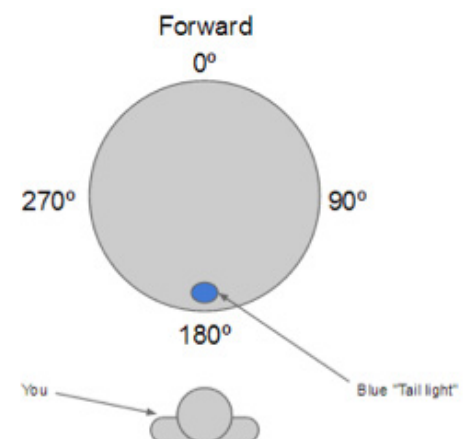
First thing you need to do is to connect the iPad to Sphero. Here's how:

1. Pick up Sphero from its charging station and tap it twice on the logo to wake it up. You may have to tap it hard. It will start flashing colors when it is awakened out of its "sleep" state.
2. On your device, make sure Bluetooth is enabled. From the home page, click on Settings at the bottom. Then choose Bluetooth.
3. You will be shown a list of Spheros. Connect to the appropriate Sphero by tapping it. You can tell which Sphero is which by the names, which relate to the colors the ball is flashing. For example, if it flashes purple, then yellow, then green, then that is ball PYG. Select the one you want. Once successfully connected, it will say "Connected".



Part 2: Aiming Sphero

Sphero has a direction built into it that it thinks of as "straight ahead". This is called the orientation. The first thing we want to do is to aim the Sphero so that the orientation is on the path we want it to go. Each Sphero has a blue light inside of it called the "taillight", which is always on the exact opposite side of the straight ahead direction. You are going to set the taillight so that it's pointing right at you when you look down the path you want Sphero to go. Then, when it goes straight ahead, it will be on that path.



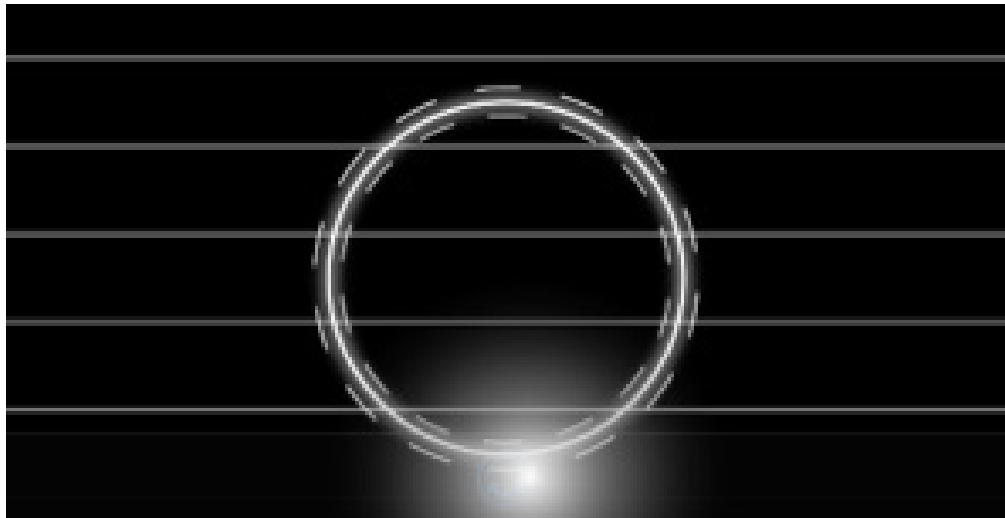
Part 2: Aiming Sphero (continued)

Follow these steps to aim the Sphero:

1. Go to the home screen and open OrbBasic.
2. Have one of you hold the Sphero and stand at the beginning of the path you will use for your experiments.
3. Now, you will aim the Sphero in that direction. Have a second member of the group use the iPad. In OrbBasic, you will see a circle with two arrows at the bottom center of the screen. Tap on it and hold it.



4. A white circle will appear. Move your finger slightly to rotate the insides of the Sphero. You will see a blue light inside the ball. Move it around until the blue light is directly facing the person holding the Sphero. This is the "taillight", and shows the direction opposite where the Sphero will move when moving straight ahead.



Part 3: Making Sphero turn purple in the air

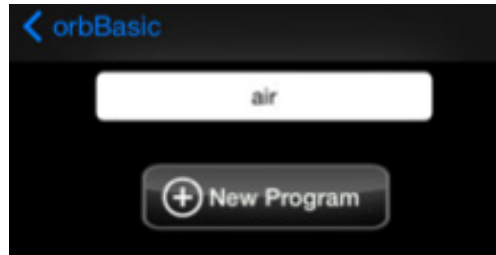
Sphero has an accelerometer inside, an accelerometer is used to measure the acceleration, or change in velocity of Sphero.

Orb Basic has a command called accelone which is a cumulative reading of the accelerations acting on Sphero at a given time in all directions (X, Y, Z). If Sphero is only being acted on by gravity (i.e. when it is in the air) the reading will be zero. Due to the fact that few things are perfect, you will need to build in a soft buffer to determine if Sphero is in fact in the air. You can set up the buffer so that if the value is less than 200 that corresponds with Sphero being in the air. You can build a very short 2 line program to change Sphero's color when it is in the air.

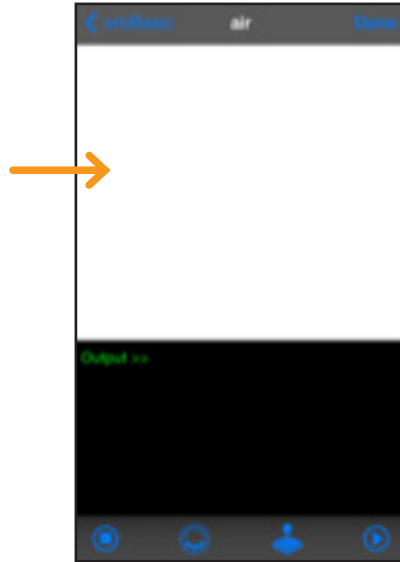
1. Tap the + button at the bottom to create a new program.



2. In the space where it says Program Name, give your program a name, then click the '+ New Program' button under it.



3. Tap in the big white space. A keyboard will appear at the bottom of the screen.



4. In the first line of code for your new program you want to check whether the accelone value is sensing that Sphero is in the air or not. To do this a conditional statement is very useful. If the accelone value is less than 200 you will want Sphero to turn purple using LEDC 5 and if the accelone value is greater than 200 you will use an else statement to keep Sphero dark. To do this you can use the line below:

```
10 if accelone < 200 then LEDC 5 else LEDC 0
```

5. Next, the program needs to go back to the beginning and run through the conditional statements again

```
20 goto 10
```

The final code will look like:

```
10 if accelone < 200 then LEDC 5 else LEDC 0
```

```
20 goto 10
```

6. Tap the Done button and then the Play button and toss Sphero in the air to see what happens.

Part 4: Collision Detection

For Part 4 you will create a macro very similar to the one you just built but it will be sensing collisions instead of whether or not Sphero is in the air. When the accelerometer value is greater than 5000 that indicates that Sphero has collided with something.

1. First you can make the default color green

```
10 LEDC 2
```

- Using accelone, detect if a collision happens
`20 if accelone > 5000 then goto 30 else goto 10`
- Now you need to set Sphero to turn red using the command below for the case where the conditional statement is true
`30 LEDC 1`
- In order to ensure it doesn't just blink red but stays red for an entire second after it hits an object the next line should be a delay
`40 delay 1000`
- Finally the program needs to jump back to beginning
`50 goto 10`
- Tap the Done button and then the Play button. Use the joystick to drive Sphero

Does your Sphero turn red when it runs into something?

Part 5: Challenge

Now that you have built a program that reacts after a collision, try to build a slightly more advanced program where Sphero drives in a straight line but if it collides with an object it changes color and rolls off in a random direction. This will combine topics from previous lessons as well. Here are some tips to guide you through building this program.

- You will need to set up a variable h, which is the heading value for sphero and to begin you will need to set your initial heading to 0.
- You may want to set an initial color for Sphero as well.
- Next have Sphero to roll in a straight line
- Program Sphero to change colors and move at a random direction if it senses a collision. (Tip: to set a random heading use the following line: `Heading rnd 359`)
- Don't forget to add a delay after this line to keep sphero moving in that direction for a half a second.
- Have your program loop back to the initial color step after this delay. Be careful with where you loop back to because you don't want to go all the way back to the first command where you set Sphero's original heading of 0.

Good Luck & Have Fun!