

[Docs](#) » [Turtlebot Quick Start Tutorials](#) » [Turtlebot First Tests](#)

---

# Turtlebot First Tests

When you purchase a new Turtlebot, you would like to make sure that its sensors and actuators work as expected and behave normally. In this tutorial, we provide you a few steps to follow to test the functionalities of your Turtlebot.



## ! Note

Learn ROS and get your ROS certificate by enrolling in the Udemy course (**Highest Rated course**) [ROS for Beginners: Basics, Motion and OpenCV](#).

Learn about ROS2: ROS Next Generation by enrolling in the Udemy course [ROS2 How To: Discover Next Generation ROS](#)

## Starting The Turtlebot

First, connect the Kobuki base and 3D sensor camera (Kinect or Asus) to the laptop and start ROS. The first thing you need to do, is to turn-on turtlebot functionalities and this is performed through the bringup operation that starts all the sensors and actuators of the Turtlebot, namely, the Kobuki base and the 3D sensor. Open a terminal and write:

```
roslaunch turtlebot_bringup minimal.launch
```

This command will start the Kobuki base of the Turtlebot. You should hear a sound when it is turned on.

### 📌 See also

For more details about Turtlebot bring up refer to [Turtlebot Bringup in ROS Wiki](#).

## Testing the bumper sensor

There are three bumper sensors in the Kobuki base: (1) one in the middle-front, (2) one in the left front, (2) one the right front. The bumper sensor helps the Turtlebot detecting collision with obstacles. To test the three bumper sensor, open a new terminal and run:

```
rostopic echo /mobile_base/events/bumper
```

You should observe the following output:

```
state: 0  
bumper: 0  
-----  
state: 0  
bumper: 0
```

This means that all bumpers are released (not pressed).

The variable `state` can take the values `0 (RELEASED)` and `1 (PRESSED)`. The variable `bumper` takes the values `0 (LEFT)`, `1 (CENTER)`, and `2 (RIGHT)`.

Now, you can check the central front bumper. Press on the central front bumper. You should observe that the turtlebot moved slightly to the back and you should observe the following output in the terminal:

```
state: 1
bumper: 1
```

When you release the center bumper, you will observe

```
state: 0
bumper: 1
```

Repeat the process for the left and right bumper sensor to make sure that their correct operation.

## Testing the cliff sensor

Cliff sensors are responsible for detecting cliffs and altitude changes when the robot is moving to prevent crashes when reaching stairs. They are located in the bottom of the Kobuki base. The behavior is very similar to the bumper sensor. There are three cliff sensors, one in the center, one in the left, and one in right side. To test the three cliff sensors, open a new terminal and run:

```
rostopic echo /mobile_base/events/cliff
```

You should observe the following output:

```
state: 0
sensor: 0
-----
state: 0
sensor: 0
```

This means that all cliff sensors detect a normal ground (no cliff).

The variable `state` can take the values `0 (FLOOR)` and `1 (CLIFF)`. The variable `sensor` takes the values `0 (LEFT)`, `1 (CENTER)`, and `2 (RIGHT)`.

Now, lift the turtlebot to the up, and you will see a similar output

```
state: 1
sensor: 0
-----
state: 1
sensor: 1
-----
state: 1
sensor: 2
```

Which means that all sensor detected a cliff as the robot is away from the ground. Put the robot back on the ground and all states should come back to 0.

## Video Demonstration

### Note

Learn ROS and get your ROS certificate by enrolling in the Udemy course [ROS for Beginners: Basics, Motion and OpenCV](#).

Learn about ROS2: ROS Next Generation by enrolling in the Udemy course [ROS2 How To: Discover Next Generation ROS](#)