

Lab 4: Roomba Control

Roomba Control



Figure 1: iRobot Create 2

In this lab we are finally going to play with the Roomba and start commanding it to move around. This will prepare you for the final project with the Roomba.

You should learn or gain experience with:

- How to command the Roomba
- How to read the sensors and react accordingly
- How to monitor the Roomba robot

WARNING: If you drop/damage the roomba, you automatically fail the lab. There are not enough robots available if we start damaging them.

WARNING: For all commands with the roomba, do not set the speed above 200 mm/sec. This is so it doesn't get away from you and cause havoc.

Authorized Resources

You may only get help within your group or from the instructor. Do not talk to other groups or other cadets about this lab.

[5 pts] Pre-lab

Hand in a copy of your code code. You should have functions for each task in the lab. Use the example code below to help you.

[25 pts] Task 1: Simple Commanding

Start off commanding the robot to move forward for 3 seconds, the backwards for 3 seconds, turn in place ~90 deg left, and finally turn in place ~90 deg right. It should approximately end up back where it started and in the same orientation.

Basic usage information can be found on: <https://pypi.python.org/pypi/pycreate2>

```
#!/usr/bin/env python
from __future__ import print_function, division
import pycreate2
import time

if __name__ == "__main__":
    # this creates the robot instance we will use
    bot = pycreate2.Create('/dev/ttyUSB0')
    bot.safe()

    # your code here
    # move forward
    time.sleep(3)
    # move backwards
    time.sleep(3)

    print('All done ... exiting :)')
```

When you have this working, show your instructor.

[25 pts] Task 2: Read Sensors in Real-Time

Now we are not going to command the robot to move anymore, but we are going to read the sensors. With the robot standing completely still, read the light bumper sensors, print the results to the screen, and use your hand to change the readings. To read the sensors, use:

```
# reading the IR sensors
sen = bot.get_sensors()
sen.light_bumper_left
sen.light_bumper_front_left
sen.light_bumper_center_left
sen.light_bumper_center_right
sen.light_bumper_front_right
sen.light_bumper_right

sen.cliff_left_signal
sen.cliff_front_left_signal
sen.cliff_front_right_signal
sen.cliff_right_signal
```

Again, for the task we are not moving the robot. Here is some code to help you get started:

```
#!/usr/bin/env python
from __future__ import print_function, division
import pycreate2
import time

if __name__ == "__main__":
```

```
# this creates the robot instance we will use
bot = pycreate2.Create('/dev/ttyUSB0')
bot.safe()

# your code here
while True: # press ctrl-c to end it
try:
    sen = bot.get_sensors()
    # print the light bumper sensor readings
    # print the cliff sensors
except KeyboardInterrupt:
    print('All done ... exiting :)
```

The point of this is to understand the values you need to avoid obstacles in the next task.

1. Light Bumpers: What are the minimum/maximum distances (in inches or cm) that the roomba can sense. Also what are their values.
2. Cliff Sensors: What are the sensor values for carpet and the masking tape?

Also, write a small program that reads the roomba's power level and reports its current battery charge. It should print out something like this:

```
Battery [mAhrs]: 1200
```

After a while, the roomba's internal battery will wear out, it is nice to know how low it is. This will probably have a bigger impact on latter class periods rather than earlier class periods.

When you have this working, show your instructor.

[25 pts] Task 3: Avoid Obstacles

Now write a program that runs and reads both the light bumpers and cliff sensors. If the light bumpers detect an obstacle, it should turn away. If the cliff sensors values decreases too much (you determined this level in the last task), the back up and turn away from the obstacle. Review the Controls lesson for ideas.

When you have this working, show your instructor.

[25 pts] Task 4: Square

Now using your obstacle avoidance routine from above, have the Roomba travel in a square that is ~2 m on each side and end up in the same spot it started in.

When you have this working, show your instructor. You still need to be able to avoid obstacles while doing this, however, if something gets in your way, you do not have to worry about ending in the same location you started in. Just complete the cube (best effort) after avoiding the obstacle.

Explain to your instructor the following questions:

- Did it work?
- How close were you?
- Why did this or why not did this work?
- How could you improve it?
- Are you able to avoid obstacles?