

CamJam EduKit Robotics - Driving and Turning

Project Driving and Turning

Description You will learn how to make your robot move in the direction you want it to.

Equipment Required

For this worksheet, you will require:

- Your robot

Going Forwards and Backwards

In the last worksheet, you saw the basics required to turn the wheels on your robot.

In this section of the worksheet, you are going to make the Robot go forwards and backwards, and make the beginnings of a robot control program.

If you recall from the last worksheet, the EduKit Motor Controller Board uses two GPIO pins to make each motor go forwards or backwards. You have to turn the correct ones on and off to make the motors, and therefore your robot, move in the direction you want it to.

To make a wheel go forward or backwards, you have to turn one pin on and the other pin off. If you turn them both off, the motor will not move. Remember the following:

- For the right-hand motor, the controller board uses pin 10 to turn the motor in the forwards direction, and pin 9 to turn it backwards.
- For the left-hand motor, pin 8 is used to turn the motor forwards, and pin 7 for backwards.

Therefore, to make your robot move forwards, you have to turn pins 10 and 8 on, and 9 and 7 off. To go backwards, you have to turn pins 9 and 7 on, and pins 10 and 8 off.

GPIO Zero has code that has been specifically written for the CamJam EduKit controller board. You therefore do not need to remember which pins to use, as GPIO Zero already knows what to do.

Code

Note: In Python, the spaces at the start of lines are important. They are Python's way of recognising code that is grouped together into functions or within other command structures, like `if`, `for` and `while`.

Create a new Python script called `4-driving.py` and type in the following code.

```
# CamJam EduKit 3 - Robotics
# Worksheet 4 - Driving and Turning

import time # Import the Time library
from gpiozero import CamJamKitRobot # Import the GPIO Zero Library CamJam
library

robot = CamJamKitRobot()

robot.stop()
```

Running the Code

You are now ready to run the code. Select the Run Module menu option, under the Run menu item. Alternatively, you can just press the F5 key.

The robot won't actually move yet!

Moving the Robot

To make the robot actually move, you need to edit the code again, and insert the following lines just before the last line (`robot.stop()`):

```
robot.forward()
time.sleep(1) # Pause for 1 second

robot.backward()
time.sleep(1)
```

Now run the code again.

This time your robot will move forward for one second, and backwards for one second, then stop.

Left and Right

The next step is to add some more code that will turn the robot left and right. Edit the code again and type in the following code after the code you added in 'Moving the Robot'.

To turn left, the right-hand motor is turned forward, and the left hand one turns backwards. To turn right, the left-hand motor is turned forward, and the right hand one turns backwards.

```
robot.left()
time.sleep(0.5) # Pause for half a second

robot.right()
time.sleep(0.5)
```

The full listing should look similar to this:

```
# CamJam EduKit 3 - Robotics
# Worksheet 4 - Driving and Turning

import time # Import the Time library
from gpiozero import CamJamKitRobot # Import the GPIO Zero Library CamJam
library

robot = CamJamKitRobot()

robot.forward()
time.sleep(1) # Pause for 1 second

robot.backward()
time.sleep(1)

robot.left()
time.sleep(0.5) # Pause for half a second

robot.right()
time.sleep(0.5)

robot.stop()
```

Once again, run the code.

Challenge

You now have a robot that will move around to your command. Experiment by writing your own code to move the robot around, changing how long they run for by changing the numbers of seconds the code pauses for using `time.sleep(X)`, where X is the number of seconds to pause. You can use decimal numbers, like 0.5 for half a second.