
BODGERS

# Last weeks code

```
from time import sleep
from gpiozero import Robot

robot = Robot(left=(22,23),right=(24,25))

def dist_to_time(dist):
    cm_per_sec = 60.0
    time = dist / cm_per_sec
    print (time)
    return time

def robot_go(dist):
    time = dist_to_time(dist)
    robot.forward()
    sleep(time)
    robot.stop()

    robot_go(100)
```

# Today

1. Time our robots going forward for 1 second
2. Time our robots going backwardfor 1 second
3. Time our robots going left for 1 second
4. Time our robots going right for 1 second
5. Use the going forward time in robot_go()
6. Copy the robot_go() function and change it for backwards, left and right
7. Test our functions on an obstacle course

```
def angle_to_time(angle):
    angle_per_sec = 175.0
    time = angle / angle_per_sec
    return time


def robot_left(angle):
    time = angle_to_time(angle)
    robot.left()
    sleep(time)
    robot.stop()
```

```
def angle_to_time(angle):
    angle_per_sec = 175.0
    time = angle / angle_per_sec
    return time

def robot_go(dist):
    time = dist_to_time(dist)
    robot.forward()
    sleep(time)
    robot.stop()

def robot_back(dist):
    time = dist_to_time(dist)
    robot.backward()
    sleep(time)
    robot.stop()

def robot_left(angle):
    time = angle_to_time(angle)
    robot.left()
    sleep(time)
    robot.stop()

def robot_right(angle):
    time = angle_to_time(angle)
    robot.right()
    sleep(time)
    robot.stop()

robot_go(100)
robot_left(90)
robot_right(90)
robot_back(100)
```

# Next week Remote Control

We will use Pygame but instead of using it to control sprites on screen we will us it to control our robot

If you can get a USB webcam bring it in and we turn our robot into remote controlled camera.