COMP105P Lab Tasks 4

Tasks

After the basic tasks from last weeks, now we move on to more interesting (and challenging) code and logic. You are encouraged to reuse as much of your code as possible in future tasks by creating header (and source) files that group together tasks that you are likely to repeat often. For example, bundle all of the logic that deals with driving a fixed distance or turning a fixed angle into one file and expose this functionality via a header file that you will include in subsequent tasks. This makes your code more modular and manageable. In order to prompt you to do this from this week on we will not be awarding credit for work submitted as a single big C file.

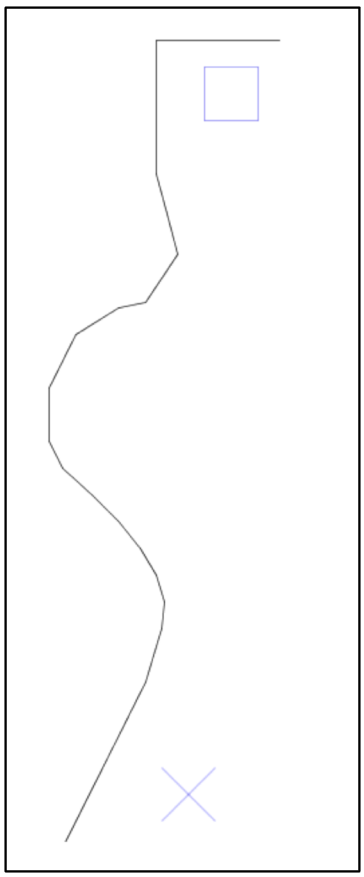

Figure 1

The task is to start from a known location and follow a curving wall until the wall suddenly turns 90 degrees. At this point, stop with the front of the robot 5cm from the wall and report the robot's position and direction, relative to the starting point.

- **Task 4.1**. Follow a curving wall on the left hand side. The starting position will be approximately 10cm from the wall. After a few meters the wall will turn 90 degrees right. Detect this turn, and stop facing the end wall with the front of the robot 5cm from the end wall (Figure 1).

- **Task 4.2**. Record the position and direction of the robot relative to its starting point in the SD card. For example, if relative to the robot's starting point and direction, the robot has moved 225cm in a direction of 10 degrees, save this information into the SD card.

Hints and Code

- Before you take on Task 4.1 make sure you are comfortable reading the sensors. Spend some time playing with them (see links below). We do not (yet) pose any restrictions as to how fast you need to follow the wall so it should be relatively simple to come up with an algorithm to do the following (we are not looking for anything very complicated).
- The following links provide sample code in order to read the sensors and use the SD card.
    - Ping sensor: http://learn.parallax.com/activitybot/build-and-test-ping-sensor-circuit
    - IR sensors: http://learn.parallax.com/activitybot/test-ir-sensor-circuits
    - SD card: http://learn.parallax.com/propeller-c-simple-devices/sd-card-data
- Follow a wall smoothly does not involve a series of straight lines and fixed able turns. You need to continuously calculate and adjust motor speeds based on the readings you take from the sensors (refer to lecture notes).
- You'll need to remember the maths of arcs and sectors. As a reminder, if you'd a sector of a circle with angle θ radians and radius r, the length of the arc is $r\theta$. As measured by the motor encoders, your robot will drive as a series of very short arcs, interspersed by very short straight lines.
- To determine the robot's position, you'll need to constantly read the motor encoders in your main loop, and use the readings to determine the robot's path. You can't just call them before you start driving and after you stop, and determine anything useful (refer to lecture notes).
- One of the challenges of the current and coming labs is to produce a working algorithm with the smallest memory footprint possible as the robot's EEPROM size is limited. In case, your code doesn't run in the EEPROM because of size limit, consider using the 2GB SD card to store data (for example to store the robot's x, y coordinates).