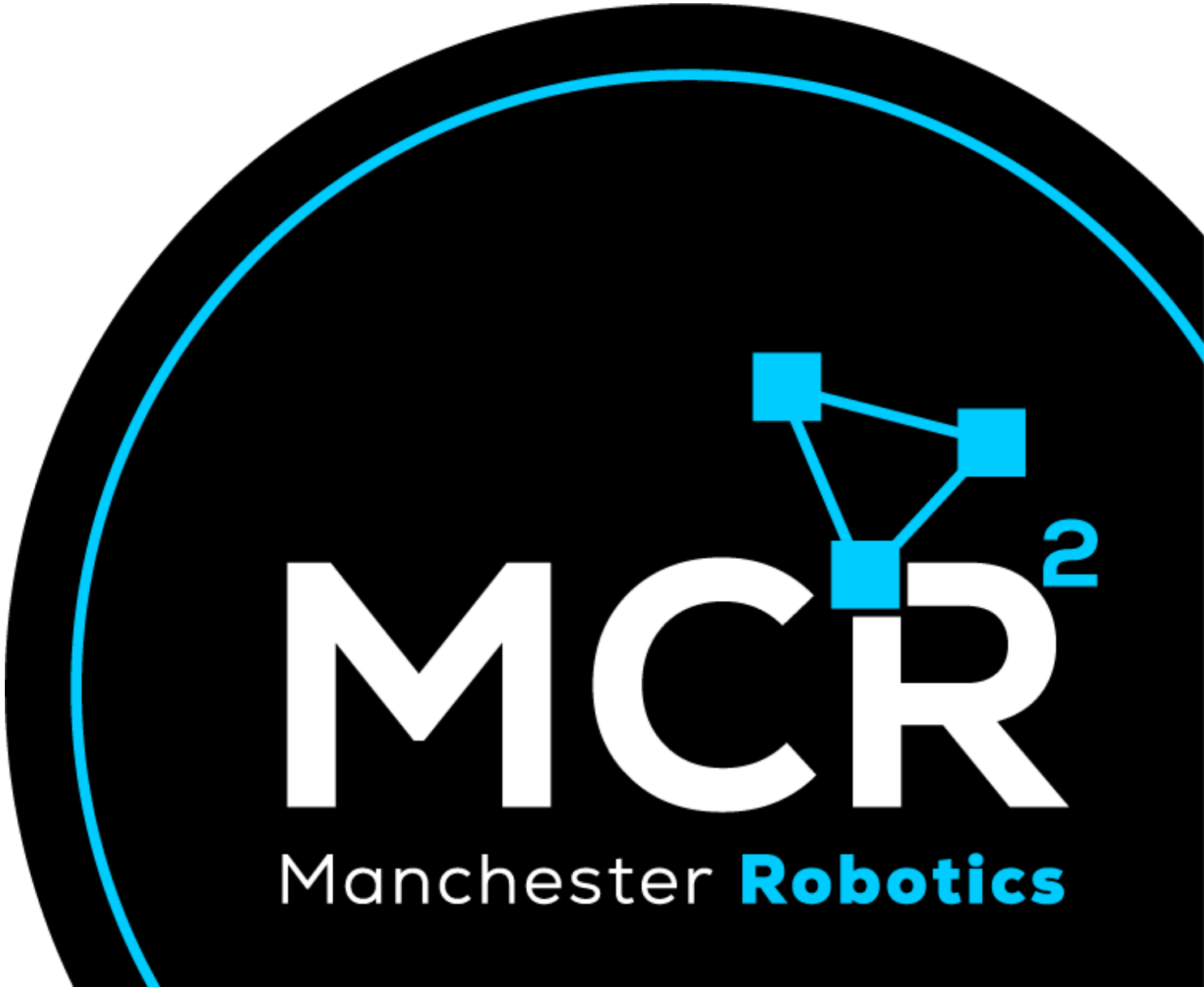
{Learn, Create, Innovate};

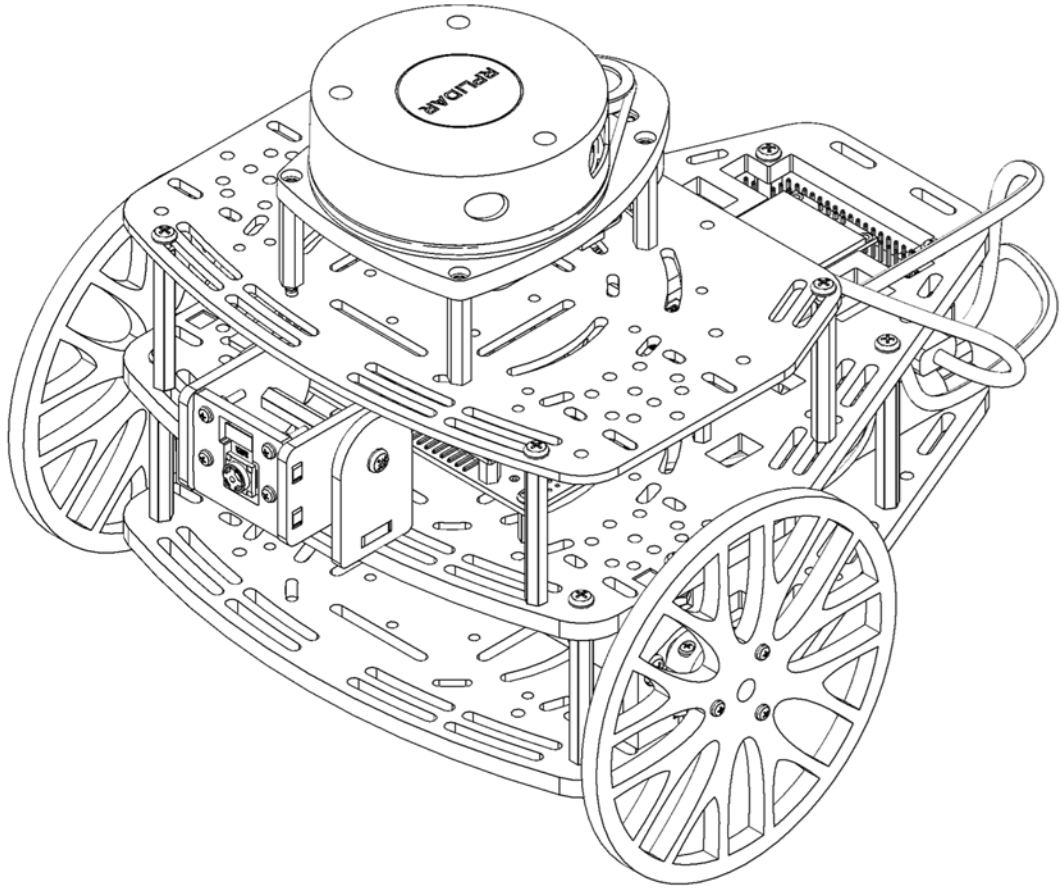
Challenges

Half Term Challenge



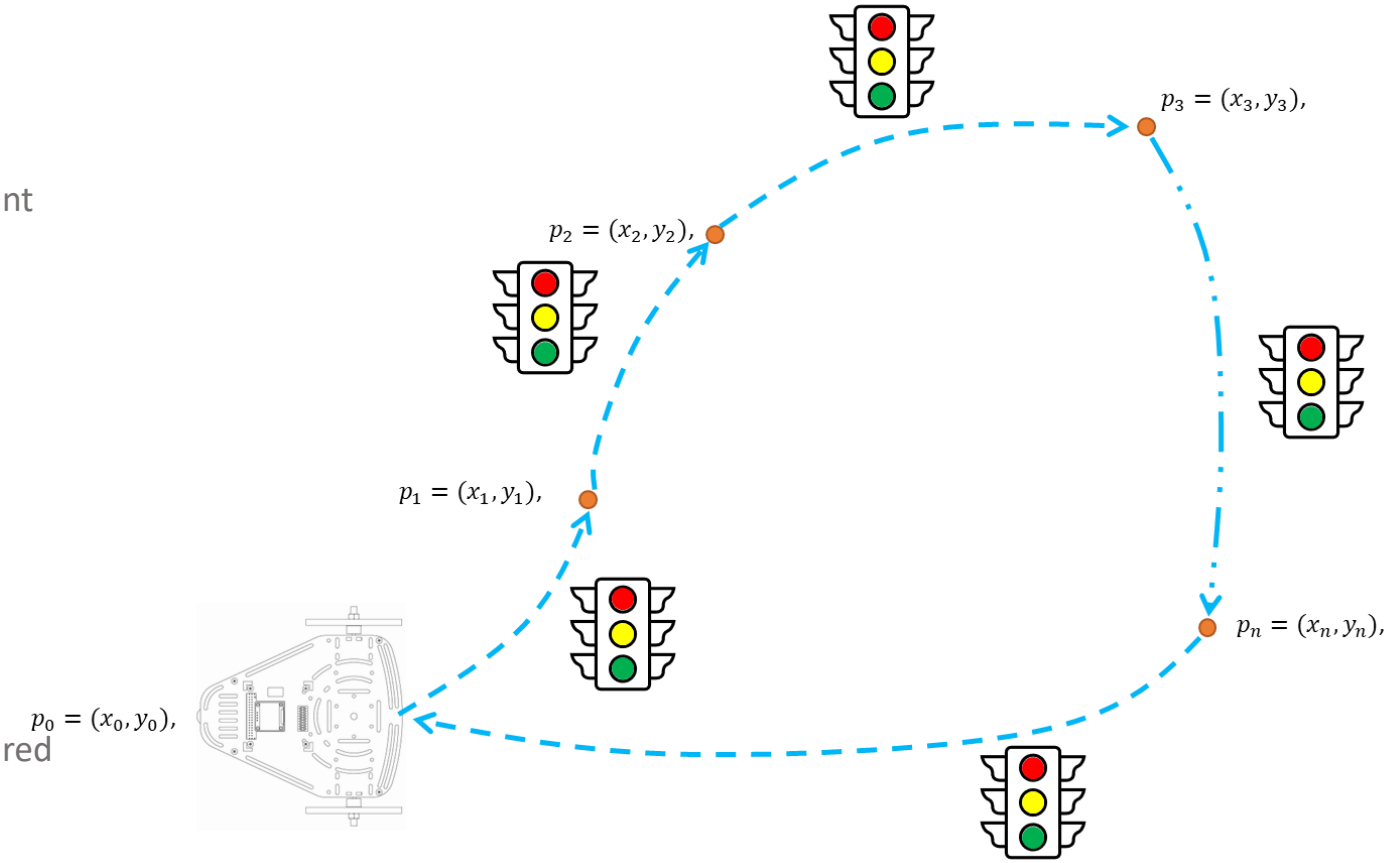
Half Term Challenge

- This challenge is intended for the student to review the concepts introduced in this week.
- This challenge aims to show the behaviour of vision systems in mobile robotics.
- This challenge will be divided in different sections.



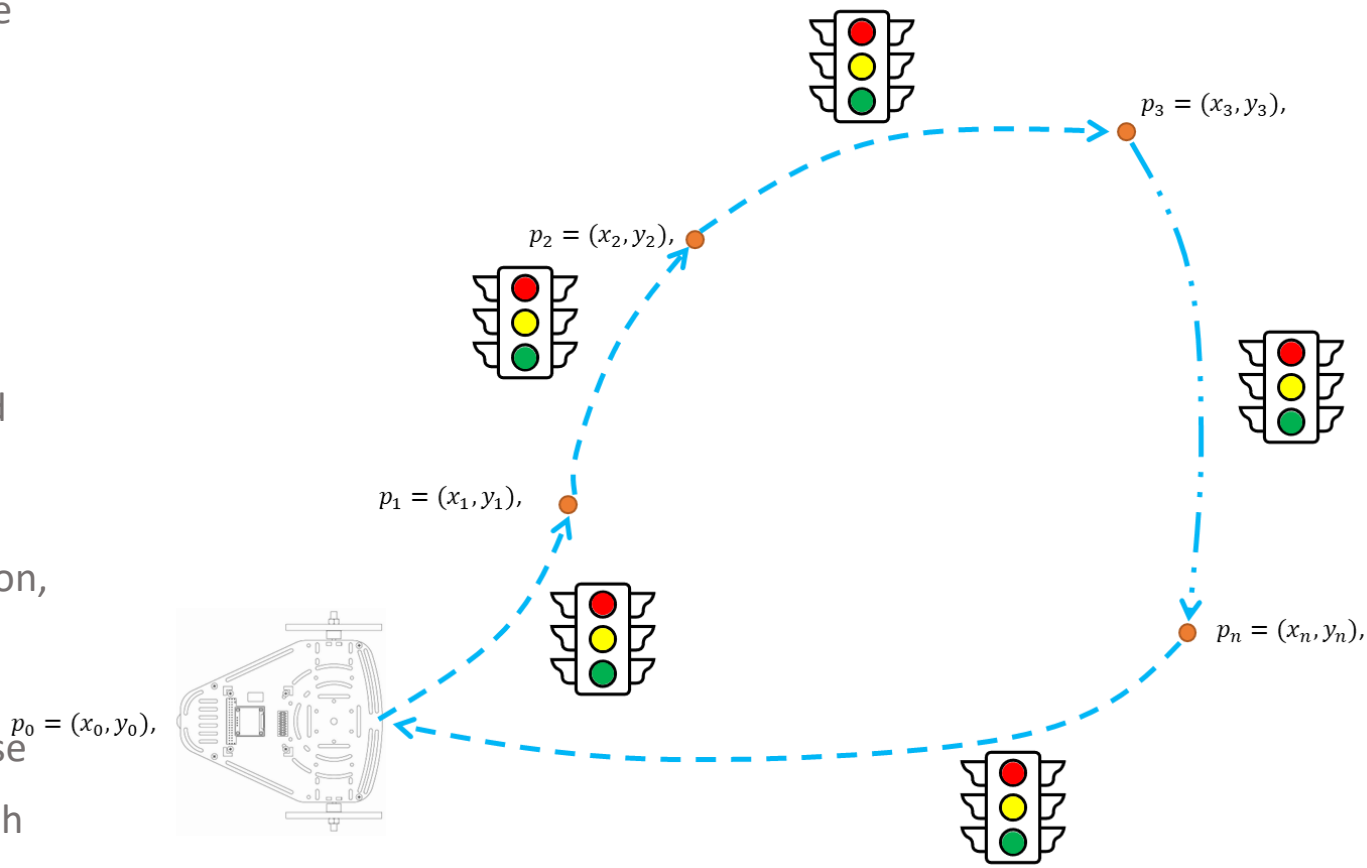
Half Term Challenge

- In this challenge, the student must use and combine the knowledge developed in previous activities.
- Add a decision-making layer to your previously developed point-to-point navigation algorithm, to detect the colour of a “traffic light”.
- The expected behaviour is:
  - Red : Stop until you see a green light.
  - Yellow: Drive slowly, until you see a Red Light to stop.
  - Green: Continue with your Path.
- Note: You must remain stopped until you see a green light (Even if the red light disappears or you are not able to detect it.)



Half Term Challenge

- The vision algorithm and closed loop controller must be **robust.**
  - The student must define what is robustness and implement strategies to achieve it with the controller.
- The vision algorithm and the controller must be tunned properly.
- The controller must take into consideration, perturbation, nonlinearities and noise.
- It is encouraged, but not required, for the student to use a config file or a parameter in the launch file to establish the goal targets such that they can be changed outside the code (not hardcoded).



Deliverables

- The students must submit a video showing their results.
- Duration: Under 3 min. (If longer, increase speed)
- Video in English
- This is an engineering report:
  - Use the third-person
  - Do not use expressions like “the controller works well”, “it's fine”, “robust”, “optimal”, etc. unless you can prove it (be serious with your results).
  - Use different metrics (maybe error-based) to analyse your system.
  - Show plots of the set points, control inputs, errors, and system output and use them to analyse the behaviour.
  - Do not conclude things like “I learned a lot”, “it was very challenging”, “everything looks fine”, etc. Conclusions based on the results, problems faced, solutions, etc.

Tasks
Brief introduction (problem to be solved, solution strategy, team tasks, etc.)
Explain how the program works (launch files, libraries made, the structure of the project, etc.). • Use the code but also use flowcharts to explain the code • Explain in this section the code, parameters, inputs and outputs of your model.
Show the results of the robot being controlled. • Try different scenarios
Explain the methodology followed to tune the controller parameters
Analysis of the robot’s behaviour. What is expected? Is the behaviour good? Why? Establish a metric to verify if the behaviour of the robot is good or not. Advantages/disadvantages of this type of Control? Problems with this type of control?
A brief set of conclusions from the task. • The conclusions should be about the practice and the theoretical aspects of the robot, not about your personal experiences.

Rules

- This is challenge **not** a class. The students are encouraged to research, improve tune explain their algorithms by themselves.
- MCR2(Manchester Robotics) Reserves the right to answer a question if it is determined that the questions contains partially or totally an answer.
- The students are welcomed to ask only about the theoretical aspect of the classed.
- No remote control or any other form of human interaction with the simulator or ROS is allowed (except at the start when launching the files).
- It is **forbidden** to use any other internet libraires with the exception of standard libraires or NumPy.
- If in doubt about libraires please ask any teaching assistant.
- Improvements to the algorithms are encouraged and may be used as long as the students provide the reasons and a detailed explanation on the improvements.
- All the students must be respectful towards each other and abide by the previously defined rules.
- Manchester robotics reserves the right to provide any form of grading. Grading and grading methodology are done by the professor in charge of the unit.