Real-time control & localisation of a rover utilising a USB Foot Pedal interface & Bluetooth low energy

THE UNIVERSITY OF QUEENSLAND AUSTRALIA

Cody Burnett, Declan McMahon, Lachlan Edwards & Riley Mosca, Supervised by Dr. Matt D'Souza

Introduction

This project focuses on developing a real-time controlled rover using a USB pedal interface connected via Bluetooth Low Energy (BLE). The rover's position will be accurately localised in real-time, and the results will be displayed on an online dashboard. The project aims to achieve the following Key Performance Indicators:

1. Control response time Target less than one second for pedal input value change to effect noticeable change to vehicle speed or direction.
2. Localisation accuracy Rover localisation should be accurate to 15cm in x and y directions, at all points on the grid.
3. Heading accuracy Rover heading (direction) relative to middle-top part of grid, is tracked continuously and accurate to 10 degrees.
4. Web dashboard response time (latency) Changing values for heading and position are updated within 2 seconds of value changing
5. Web dashboard refresh rate (throughput) Web dashboard values update at rate of 1Hz during operation.
6. Control 'feel' Controls feel responsive, and vehicle moves at speeds proportional to pedal input level. Controls are accurate and precise.

System Overview

The implementation of this project (*Figure 2*) involved the utilisation of various micro-controllers, devices, sensors, peripherals, and clients. See*Figure 1*.

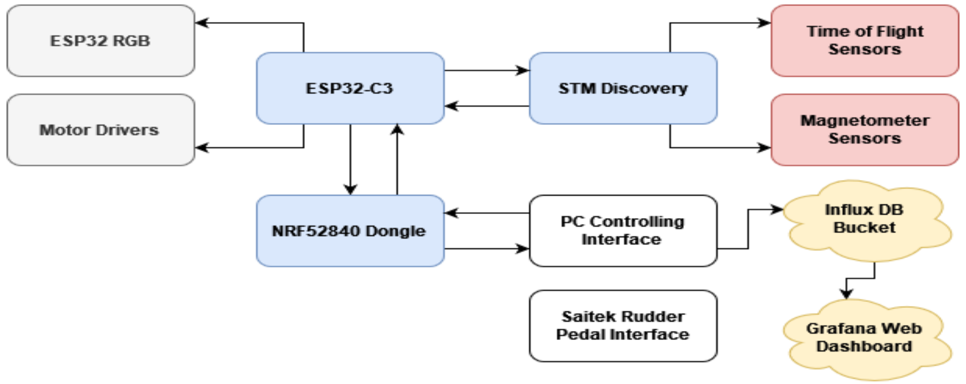

Figure 1. System Overview

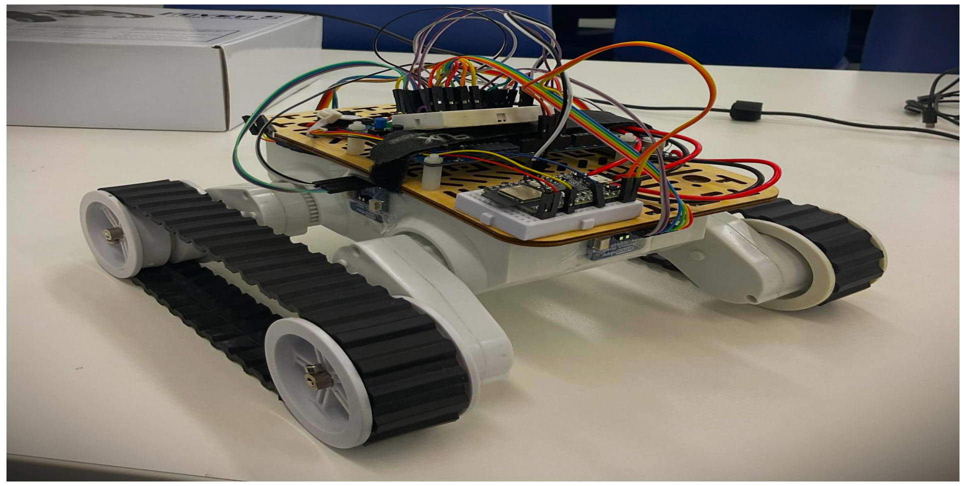

Figure 2. Remote-Controlled Rover

Methodology

The successful delivery of this product relied heavily on the implementation of agile development methodologies and modular design abstraction. These practices enabled a consistent development process with rigorous testing and review. The use of independent software modules reduced cohesion and allowed for simultaneous development of software packages for all devices and peripherals. The design methodology of this project is detailed below:

1. System requirements analysis
2. High level system architecture design
3. Sensor & driver module development
4. Position & Heading calculation Development for sensor fusion and localisation
5. NRF52840, ESP32-C3 STM configuration
6. Development of UART & BLE communication & standards
7. PC Client & Web server interface development
8. Product testing, integration & validation
9. Product deployment & monitoring

Design Limitations

1. ToF Sensor Range, the Time of Flight (ToF) sensor is limited to a 4 metre range, which is sufficient for this application, however, if scaling the project, ToF sensors with longer ranges may be necessary.
2. Position Tracking The position tracking of the rover is conditional on a fixed boundary around the rover..
3. Localisation of the rover requires component values for the x and y co-ordinates, which requires the angle of the rover to be captured during run-time.
4. The TOF sensors used for this project (VL53L1X) reduce performance as the ambient lighting of the room increases. This may pose limitations to localisation if operating the rover in bright conditions.
5. Magnetic Field Interference The magnetometer readings used to determine the angle of the rover are heavily influenced by the magnetic field generated

Field Deployment Plan

The Rover is to be deployed in a $4m^2$ perimeter as shown in *Figure 3*. The placement of the rover is in reference to the origin, with its orientation configured at run-time. The Rover must be connected to power - enabling the motors and micro-controllers.

The *Saitek pedals* & the *NRF52840 Dongle* must be plugged into the *Host PC*, where the script can be run. This script will then communicate with the *ESP32-C3* to allow full control of the Rover.

Due to the modular nature of this project, additional hardware & software modules can be added or configured to scale and enhance the current design. The current operating region is fixed, therefore deployment in larger/ open regions will require additional settings and localisation methods to work effectively.

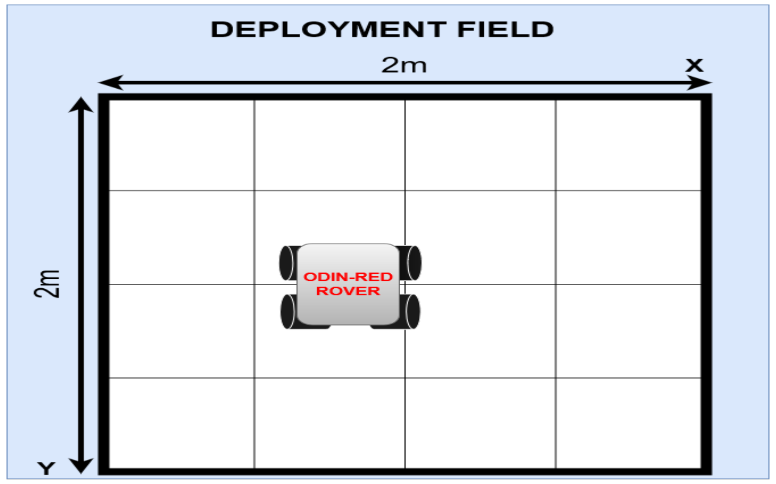

Figure 3. Rover deployment configuration

Results

The Rover successfully responded to pedal controls, and navigated within the $4m^2$ testing area. The positional data and heading was accurate with the ESP32 providing responsive updates to rover position. The position values were plotted, for a number of movements in *Figure 4*. This shows tabular, and visual plotting of the position at near real-time.

Figure 4. Grafana Dashboard Visualisation

Conclusions

1. **Rover Control - Latency:** The rover responds almost immediately to pedal controls, but is prone to higher latencies with weak BLE connections or interruptions.
2. **Localisation Accuracy:** Position data is extremely accurate, with component x & y values within 5cm of actual rover position.
3. **Heading Accuracy:** Angular heading is extremely accuracte, and is in most cases within 5 degrees from actual heading.
4. **Dashboard - Throughput & Latency:** InfluxDB Data is received as soon as it is processed, and Grafana extracts this published values immediately. Noticeable delays in position tracking is due to Grafana's 5s refresh & millisecond delays in processing on Python.
5. **Control 'feel':** The pedal data is extracted & sent to the ESP32-C3 immediately, and pedal control is proportionate to data sent.