**Milestone 4 Report**

Authors: Kian Frassek, Cameron Clark and Thiyashan Pillay

# Introduction:

A line follower robot is a common robotic task where a robot with a simple drive system, sensors and a microcontroller are tasked to follow a line and sometimes complete tasked along the way. In this iteration of a line follower robot design challenge, there is a treasure hunt taking place. The goal is for the robot to take a twisting path with intersections and dead ends, stopping at all the specified measurement lines to measure the distance to the treasure objects using an ultrasonic sensor, and finally stopping at a specified black rectangle as fast as possible.

The provided equipment includes the chassis (the structure, wheels, motors, and motor drivers), the sensors (five-line sensors, two motor rotation sensors, one ultrasonic sensor), the batteries (with power switches) and finally the microcontroller. Our goal is to regulate the two 18650 batteries to a stable 5V to power the onboard equipment as well as the coding the brain of the operation: the Arduino Nano.

This document is an in-depth analysis of the evaluation of the performance of the robot and whether or not it has met the required objectives set out in the beginning.

# Summary of Acceptance Test Procedure:

In milestone 1, there was a list of merits on which the final design was tested against. The table is shown below:

|  |  |
| --- | --- |
| Power On | Verification that the robot powers on and initializes successfully. |
| Interconnectivity | Ensuring that all subsystems and components communicate effectively with each other. |
| Movement | Validating that the robot can move autonomously as intended. |
| Speed and turning control | Verifying precise control over speed and turning during navigation. |
| Line following | Confirming that the robot can accurately follow a line on the designated path. |
| Intersection resolution | Ensuring that the robot can correctly resolve intersections and continue its path accordingly. |
| Path mapping | Validation of the robot's ability to map paths and keep track of the routes taken during navigation. |
| Stop and measure distances | Verifying that the robot can halt at specific measuring points and accurately measure distances to objects. |
| Termination | Confirmation that the robot terminates its movement when it reaches the designated stop point or the end of the maze. |

# System level design: Overview of design concepts and justification for choices.

# Electrical design:

## Power calculations and Power supply design and schematics (including safety)

## Microcontroller resource allocation

## Design and Schematics of circuits with Vero/PCB layout

# Motion control design, implementation, and results Include.

## Pseudocode functions

## UML or flow diagrams of code or MATLAB Simulink layout

## Results and justifications

# Line sensing design, implementation, and results Include.

## Pseudocode functions

## 

## UML or flow diagrams of code or MATLAB Simulink layout

## Results and justifications

# Treasure Maze solving algorithm design, implementation, and results.

## The modular design of software down to function calls and including interrupt / real-time structure.

## UML diagrams (flow diagram of code)

## Pseudocode functions

Don’t have MATLAB functions, so there is no pseudocode.

# Conclusion