MANCHESTER 1824
The University of Manchester

Department of Electrical and Electronic Engineering

Embedded Systems Project 2022-23

FINAL REPORT

Group Number: 12

Group members:	ID Number	I confirm that this is the group's own work.
Andrew Slater	10908845	✔
Kynard Tey Kai Min	10915992	✔
Tingxuan Wang	11174443	✔
Aqil Kamaruddin	10922046	✔
Fotis Kougionas	10727014	✔

Tutor: Andrew Forsyth

Date: 11/05/2023

Contents

1. Introduction

This report describes the complete racing system developed by the team to deliver an autonomous white line sensing buggy with external controls provided by a Bluetooth module. The complete racing system is formed by the merging of various technical components including mechanical components, electronic hardware, control algorithms and software systems; all of which are designed, built, tested, and improved in stages along the course of the embedded systems project.

The executive summary provides an overview of the key facts and findings contained in this document, provided in a non-technical style.

The final system components section shows a detailed documentation of the technical design of the racing system and final improvements made to further better its performance. In addition to the technical aspect of the project, effective project planning is also crucial for the team to work together efficiently to deliver the deliverables within the time constraints that are given.

The team organisation and planning sections show the management aspects of the project and various planning done to better improve teamwork. It highlights the objectives of the project, providing a clear vision to the team so that the project may be broken down into different stages and tasks.

The budget vs outturn section describes the spendings used to develop the racing system of the buggy and explores the marketability of buggy.

The analysis of the heats section describes the analysis of the buggy's performance under a test in real-world conditions and the preparations undertaken by the team to guarantee the success of the buggy to the utmost of their abilities.

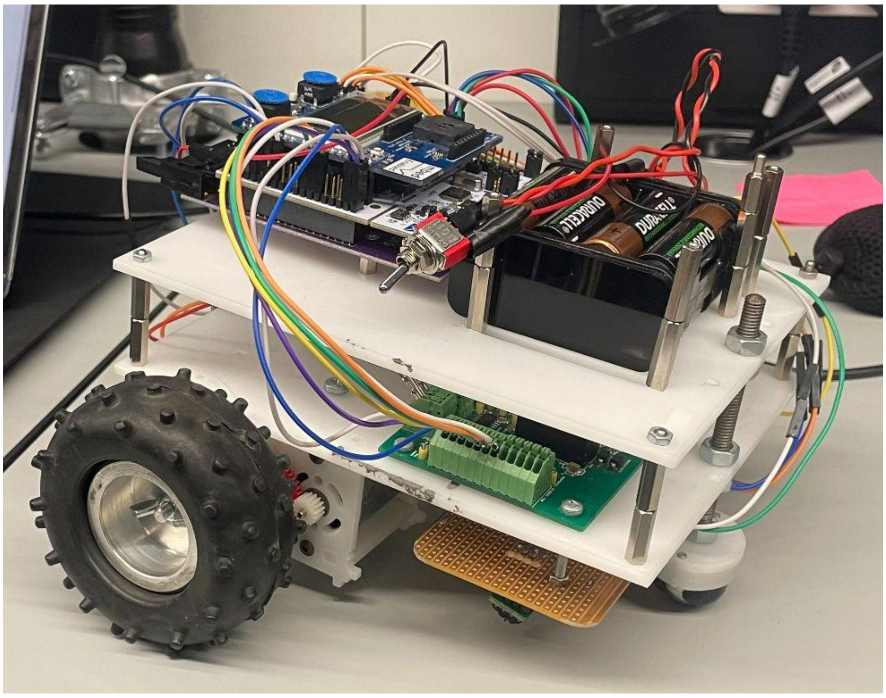

Figure 1.1: Diagonal profile of buggy

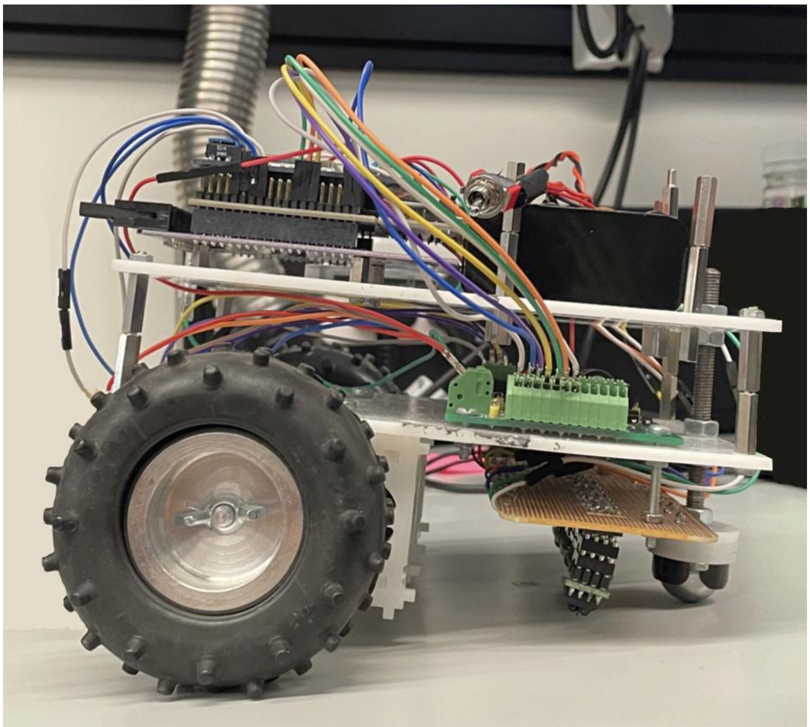

Figure 1.2: Side profile of buggy

2. Final System Components Summary

The final racing system of the buggy was made up of mechanical, electronic, control and software components which allowed the buggy to move on the given track steadily by following the white line accurately. This includes overcoming numerous obstacles along the track such as going up and down the slope and turning the buggy around at the turnaround point remotely using Bluetooth data transmission. This system for the buggy was built in preparation for the fourth technical demonstration, the heats.

2.1 Mechanical Components

At the final stage of the buggy's construction and completion, the main mechanical components used in the buggy's design consisted of the chassis, gearboxes, and motors.

The first main mechanical component, the chassis, remained almost the same throughout the entire project. The main reason for this is that the chassis fit within the size restrictions provided in the ESP procedures handbook [1], whilst also providing more than sufficient room to house all components required for the buggy to function, such as the microcontroller or motor driver board. This chassis consisted of a two-layer design, in which the lower layer is equal in size to the upper layer. This proved effective as it provided enough space to be able to change the position of any of the major components situated on the chassis, in case the team wanted to alter the buggy's centre of mass through repositioning these components. By using two layers of equal height, it also allowed the stress generated by the weight of the other major components to be spread more evenly, increasing the robustness of the buggy, and reducing the chance of chassis failure . The changes made to the chassis were relatively minor, consisting of extra holes being added to the upper layer and lower layer of the chassis. This was done in order to be able to create a battery holder using spacers to keep the batteries securely placed on the buggy chassis. Holes also had to be drilled in the lower part of the chassis to be able to fit the gearbox correctly as a hole which was included in the CAD file was originally missing on the chassis.

Another important mechanical component for the buggy was the gearbox, which maintained the same gear ratio throughout the duration of the project. This is because the gear ratio chosen for the gearbox provided a mix of both speed and torque, guaranteeing the buggy had enough torque to climb the slope, but also provide a sufficient amount of speed to be able to navigate the rest of the track as required. The gearbox consisted of three gears with a specific separation between the gears. The choice of gear ratio and separation between the gears proved to be effective as the buggy successfully climbed the slope during the heats, whilst also containing the ability to move quickly if needed. A change made to the gearbox was the replacement of the left motor's gearbox before TD3, which was made as the gears in this gearbox didn't rotate as smoothly as the other gearbox. The gearbox and motor were taken to the correct faculty members who replaced the gearbox with a perfectly working one.

A third important mechanical component is the motors. These motors themselves could not be changed during the project to provide a different speed or torque, however one was replaced as part of the gearbox replacement.

2.2 Electronic Components

The main electronic components used in the final rendition of the buggy consists of the motor driver board, Bluetooth module and microcontroller.

The motor driver board remained unchanged throughout the project as it proved to be reliable throughout the four technical demonstrations. The settings for the motor driver board were also kept the same throughout the project as the original settings were deemed to be sufficient for the four demonstrations. The settings for the driver board consisted of a PWM signal, direction signal and bipolar setting signal for both motors, along with an enable setting. The choice of bipolar mode worked well as it provided easier to make and quicker changes to the speed of the motors using the microcontroller.

The Bluetooth module also remained the same throughout the project as it was the only module available. This component did not work well as the connection between it and the group's phones was extremely hard to obtain due to the large amount of Bluetooth connections available during technical demonstrations. When attempting to send a Bluetooth command, the Bluetooth module also proved to be unreliable as it randomly disconnected during the heats, causing a failed run and consequently removing the chance to be able to test and change the turnaround code for the buggy.

The microcontroller remained the same throughout the project, however the code included on the microcontroller was changed frequently to meet the demand of each technical demonstration. The final version of this code consisted of all the control algorithms described in section 2.3 of this report, along with the other algorithms described in section 2.4 of this report. This worked fairly effectively as it allowed the previous code and experience gained to be used in the final version of the buggy, reducing the time required for development.

2.3 Control Components

The control components of the buggy were designed meticulously and tested repeatedly as they formed the foundation of the movement of the buggy. Using multiple testing phases, the final control system was improved and tuned to better suit the design criteria. The control system was programmed to react to various scenarios present on the track such as moving in a straight line, performing left and right turns smoothly, going up and down a set slope in a controlled manner, turning around without touching the barriers and coming to a complete stop automatically at the end of the track. It performed this by taking inputs from the sensor system and using these inputs in a fine-tuned PID algorithm to calculate an error value that would then be applied to the duty cycle of both wheels separately to allow the buggy to react to the situation on the track. The constants for this PID controller were calculated using the formulas provided by the technical handbook [2]. Additional control components used include the speed control algorithm and the turnaround algorithm.

The PID algorithm is responsible for allowing the buggy to turn left and right smoothly. The buggy performs a turn around a corner by varying the speed of its wheels, causing a speed difference between the wheels to occur, which ultimately leads to a turn in the direction of the slower wheel. The speed of an individual wheel is controlled by its respective duty cycle produced by a PWM signal output from the microcontroller and can be controlled by applying a proportional, integral, and differential constant. The PID algorithm calculates the error which is applied to the duty cycles. The proportional constant (k_p) determines how much the buggy tries to correct the difference between the buggy's location and the white line. The proportional constant was initially set to a low value of 0.15 as the buggy's sensitivity to the difference between its position and the white line is lowered, reducing the chance of oversteering when taking a corner. Taking into account the potential fall in battery voltage in the heats, kp was increased to a value of 0.17 to increase the sharpness of the correctional turns to allow the buggy to follow the white line with smaller deviation from the line. The integral constant (k_i) eliminates the steady-state error of the buggy, allowing the centre of the buggy to align with the line. This prevented scenarios where the buggy detects the turn too late which causes it to overshoot the corner. ki was set to a value of 0.02 after many tests. The differential constant (k_d) allows for damping to be applied to the alignment of the buggy allowing it to achieve the desired value faster. k_d was set to a low value of 0.01, as setting it too high would cause the buggy to overshoot when trying to realign, making the buggy sway when moving forwards or turning.

The buggy moves in a straight line with a pre-set duty cycle of 72.5% - the duty cycle was not set too high as setting the duty cycle too high unnecessarily would hinder its performance. An example of these hindrances would be causing the buggy to overshoot the turning around a corner, draining the energy level of the battery too quickly, ultimately causing the buggy to fail at later sections. The draining of the voltage is a serious concern, as having a low battery voltage would result in a reduced speed despite having the same duty cycle. A speed control algorithm was then designed using a set of threshold-bounded conditions alongside the speed sensing capability of the encoder present on the wheels in order to counter this. The speed control algorithm essentially increases or decreases the duty cycle of the wheels by a specified value depending on the speed of the buggy. The speed control algorithm is also the main component designed to allow the buggy to travel up and down a set slope in a controlled manner. The speed control algorithm was preferred over a PID implementation up the slope due to it having a higher success rate. This is due to the increase in duty cycle by a predetermined specified value as compared to the calculated error obtained by the PID algorithm, as the error calculated by the PID may not be sufficient for the buggy to go up the slope due to lowered battery voltage. The speed control algorithm sets the duty cycle to 90%, which is often much higher than the duty cycle adjusted by the PID algorithm and is therefore less susceptible to failure due to low battery voltage. The turnaround is another aspect of the control system designed, and is controlled by a Bluetooth signal. The turnaround of the buggy was performed by applying a set duty cycle of 70% in opposite direction for both wheels causing the buggy to rotate around its axis of origin when the Bluetooth signal was received. The buggy would then wait for a set amount of time (3 s) before it returned to a normal driving state. Lastly, the buggy's automatic stop was implemented by programming a counter which increments whenever there was no line detected by the buggy. After the counter reaches a value of 50, indicating a time of 0.5 s has passed without any line detected, the duty would be set to 50% for both wheels, which stops the wheels, allowing the buggy to come to a complete stop. A time of 0.5 s is also more than sufficient for the buggy to pass any potential line breaks in the track without misidentifying it as the end of the track.

2.4 Software Components

Due to the experience gained from the technical demonstrations, the final set of software components incorporates a wide range of coding sections. For the final version of the buggy, the components required to complete the heats included the algorithms described in section 2.3 of this report, along with code used to respond to these control algorithms, such as the code used to power the motors, interact with the Bluetooth module and the code used to set up the buggy when started. The effectiveness of this code was varied, as some areas of the code related to the output to the motors worked as expected and efficiently, however some code such as the speed control and turnaround code were shown to be less effective during the final demonstration. The code for the Bluetooth command was simple and quick to develop, making it work well, along with the start-up code which was also easy to implement and fast to develop. Apart from the bluetooth connectivity, the code created for the buggy start-up and bluetooth receiving code performed well during testing and also managed to work during the heats.

3. Team Organisation and Planning

3.1 Project Objectives

The main objective of the project was to build a robust and fully automated buggy that was capable of following a white line smoothly while overcoming constraints along a provided track.

In order to complete this critical and most important objective, several objectives regarding designing the buggy and building it must have been completed, specifically the two design reports in semester 1, and the proposal document and four technical demonstrations in semester 2.

This would require a system of components that would contribute to the main objective. These components were built throughout phases of development of the buggy divided up by the four technical demonstrations. Each technical demonstration had its own objectives and served as milestones, tackling a specific component of the entire system.

The first phase was the design of the mechanical components of the complete system. The objectives consisted of choosing the best gearbox that complements the motor of the buggy and controlling the duty cycle of the motor. These objectives were completed during design report 1 and technical demonstration 1, respectively. The objectives of the first design report consisted of the correct selection of the gearbox, in which gearbox 2 was calculated to be the best gearbox [3]. The objectives of the first technical demonstration were the complete assembly of the chassis, the successful use of PWM signals to control the duty cycles of the wheels, and the ability to sense the speed of the wheels using encoders [1].

The second phase was the design of the electronic components of the complete system. The objectives consisted of the creation of a sensor board capable of reading the white line, as well as, equipping the buggy with a Bluetooth module that can receive commands.These objectives were completed during design report 2 and technical demonstration 2 respectively. The objectives of the second design report consisted of the selection of the sensor that performs the best according to the needs of the team and the creation of a PCB that would house the selected sensors. The sensor selected was the TCRT5000, as it had the best performance when compared to other sensors [4]. The objectives of the second technical demonstration were the creation of a complete sensor circuit board that is capable of sending inputs to a microcontroller, as well as, the connection of the Bluetooth module to the microcontroller and sending an instruction to it via a phone [1].

The third phase was the design of the control components of the complete system. The objectives consisted of the integration of the mechanical and electronic aspect of the complete system and the creation of a set of algorithms that would allow the buggy to respond to the obstacles present in the track as planned by the team. These objectives were completed during the third technical demonstration.

The final phase of the project was the improvement of the various components of the buggy. The objective of this phase was the successful navigation of the buggy through a track with obstacles in a single uninterrupted run. These objectives were completed to a certain extent during the Heats.

At the beginning of the project, the team's main objective was to build a fully autonomous buggy capable of following a continuous white line smoothly at a fast speed. This would allow the team to join the race competing with other team's buggies. However, this was no longer the case during the third phase of building the buggy, as the team realised the difficulties in the integration of the various components of the system into one complete system. The team faced a difficult task of fine tuning the parameters of the control algorithm and was forced to sacrifice the speed of the buggy to allow for a more consistent performance. This decision shifted the end-goal of the team to one that aimed to produce a functional autonomous buggy but was not capable of competing with the other teams. This decision also fit the aim of the project better as it contributed to the main goal of creating a robust and fully autonomous vehicle that prioritises consistency and was sensitive to the changes in its surroundings.

The main objectives of the team for semester one did not change during the project. Team members had to attend the labs and gather information for the design reports.

During the second semester, changes to some project objectives were made. At first, the turnaround for the buggy was designed to be line sensor based. Due to time constraints, the turnaround was redesigned to be time based. Although this did affect the reliability of our buggy, this change was crucial because we were able to implement the turnaround in time, before TD3. This change was not very beneficial to the overall aim of the buggy project as the turnaround was hard to time correctly. Another objective that was changed before TD3 was the speed control. The speed control was coded to be variable but due to time efficiency it was changed to discrete speed levels. This sped up the development of the buggy but reduced its reliability on the track, however it was proven to be worth the trade-off as the buggy successfully completed the uphill and downhill portions of the heats track.

The initial objective of the team regarding the attachment of the sensors on the buggy was to use a PCB. Unfortunately, the PCBs the team received were either entirely or partly faulty, rendering them useless. After soldering the components on the boards, they couldn't be used. Overall, due to connection (wrong schematic design, faulty connections, damaged board) and efficiency reasons the team decided to switch to a stripboard for the final two technical demonstrations, due to its higher reliability.

3.2 The Gantt Chart

Throughout the embedded systems project, the Gantt chart maintained a crucial role of acting as a tool to provide structure for the team's work schedule, allowing the team to track the progress of the project at any point in time, allowing for changes when necessary.

Major changes to the Gantt chart occurred after every significant deliverable was delivered, as additional detail was added to the Gantt chart due to the group's increased focus on the next set of milestones.

When the Gantt chart was created initially in semester 1 as part of the Engineering Management course unit, the Gantt chart mainly focussed on semester 1, with reduced detail for semester 2 activities as these activities didn't require a high level of detail due to their reduced relevance compared to the semester 1 activities during the first semester. This drastically changed in semester 2 as more detail was included for each technical demonstration, such as splitting the demonstrations into programming, assembly and testing components, further increasing the detail of each demonstration by including the specific requirements set out in each demonstration as subtasks within the programming and assembly sections of each technical demonstration, as shown in the team's the proposal document [5]. This matched the detail of semester 1 in which the specific sections of each design report were added and tracked individually as part of the design report deliverables. In addition to these changes, the Gantt chart was changed after each milestone was met, to include the progress the team has made on the specific subtasks for each major deliverable.

Most deliverables went to plan, as the team received high marks in all design reports and the proposal document, whilst also receiving high marks in the first three technical demonstrations. The fourth technical demonstration didn't go fully to plan as the buggy could not complete all sections of the track, however most sections could be completed, meaning the fourth demonstration still went mostly to plan. Most deliverables were achieved, as shown by the data gathered in the two design reports and high marks received throughout the project.

Very careful planning was conducted before reports and technical demonstrations, to ensure all deliverables were achieved on time. Before each report, team members always assigned a draft date (7-5 days before the submission date). Once the final version was completed, cross-checking was conducted from each team member to make sure each section of the report adhered to the requirements of the marking scheme. After the refining of the final version, the report would be submitted. For the technical demonstrations, all team members attended the labs and went through the checklist one-by-one to ensure everything was prepared before the demonstration. The team members made sure to revise their knowledge regarding the buggy in anticipation of the questions on the last demonstration. If there was any ambiguity regarding the function of the buggy, team members discussed and clarified these questions before each demonstration making sure that all members were up to date with any changes made to it.

3.3 Group Difficulties and Resolution

The main difficulty our team struggled with was the distribution of the workload regarding hardware and software. Initially, the team had a 3-2 Split with 3 members on Hardware and 2 members on software. During the first 5 weeks of the second semester, the team worked efficiently with this split, mainly due to the buggy assembly requirements. From weeks 6 and onwards, once the buggy was partly assembled and no major adjustments were needed, the hardware members weren't able to contribute equally for the preparations of technical demonstrations. Shortly after the completion of TD1, the team decided to change split to a 2-3 with 2 members on hardware and 3 members on software, as more programming was required for the remaining 3 technical demonstrations. The hardware members focused on soldering components onto the PCB and stripboard, while the software members were able to complete coding for the buggy in Mbed and design the necessary system diagrams (schematic, circuit layout, wiring diagrams). During the final weeks, once the buggy was fully assembled, workload between team members was divided between testing the buggy on the track and report writing.

For the buggy assembly, the team initially had decided to attach the battery pack with Velcro straps or a bent metal sheet inserted through the two slits in the top layer of the chassis. Both of those options were either unable to be implemented correctly or would require too much time in terms of designing. The team ultimately decided to stabilise the battery using 4 bolts inserted through the slits that would hold the battery pack in place. A hatch was implemented temporarily to prevent the battery pack from falling off the buggy during test runs on the track. After further discussion, it was decided that since the buggy didn't wobble on the test track, there wasn't any possibility that the battery pack would fall out of the buggy. The removal of the hatch and the design with the bolts in general, made the retrieval and insertion of the battery pack very easy when the batteries needed to be charged. All that needed to be done to retrieve the battery pack was loosen 2 of the 4 bolts holding it in place.

The team also struggled with some components. The chassis of the buggy was designed incorrectly at first as the hole for the castor wheel on both layers and the hole for one of the insertions of the gearbox on the bottom layer were both missing. The batteries the team received were not able to charge properly, and discharged very fast. One of the gears on the given gearbox was not spinning properly and was occasionally jamming during testing. One of the two motors provided had its polarity reversed, which seemed to be a manufacturing issue. Problems with the batteries, the chassis and the gearbox, were resolved through visiting the workshop. The necessary holes were drilled for the chassis, the batteries were replaced and the gears on the gearbox were adjusted to rotate properly.

The team faced some difficulties with the technical demonstrations. Regarding TD2, thorough testing for the BLE module was conducted and the microcontroller response to the command was successful, but the team faced connectivity issues during the demonstration which led to the failure of a task. Before TD3, the team members made the necessary changes and came up with an effective solution regarding connectivity of the BLE module. The turnaround of the buggy was successful during testing, but the team faced timing difficulties during TD3. Another difficulty the team faced was the right-hand turn. The height of the very right sensor affected its sensitivity which occasionally caused the buggy to not turn fast enough and go off track. The duty cycle was changed which improved the buggy's performance but it did not fully resolve the turning problem. In addition, the team struggled with the turnaround. The microcontroller successfully received the turnaround command, but the code made it so that sometimes it would turn indefinitely and other times it could not make the 180° turn but instead stopped either too early or too late, causing the buggy to crash. Lastly, the team also struggled with speed control. The buggy was not slowing down enough going down the slope, causing it to descend at a fast speed and going off track. This problem was addressed before TD4, where the team successfully completed this part of the track.

3.4 Teamwork Improvements

Better Leadership, task distribution and planning are all things that could have been improved in the team if the team were to repeat the project again. The team did not assign a project leader up until the last weeks of the first semester. The roles of the team members were not decided initially, which added ambiguity in terms of decision making and collaboration. Having a project leader early on to define and delegate workload among team members would have improved teamwork, decision making and task efficiency drastically.

4. Budget Vs Outturn

4.1 Budget & Outturn

A budget of £40 was allocated to the team for the purchases of additional items that would be needed to improve the buggy's performance. Only £3.60 was spent to purchase an additional five TCRT5000 sensors, for testing and to act as spare components for the team. The sensors could be reused for the sensor and the stripboard. The £36.40 then remained unchanged since the proposal stage through the entirety of the project.

The performance of the buggy during the heats was sufficient and the vast majority of the objectives were completed. However, there were flaws in the buggy and it could have been further improved with some special component purchases. An example of this would be a voltage indicator to measure the voltage level of the battery, as majority of the design flaws of the buggy were caused by the inconsistencies of the battery voltage. The cost of such a product was less than £10 and was well within the budget of the team. Such a purchase would have greatly improved the design and testing process of the buggy, as a lot of time was spent tuning the parameters of the control algorithm due to the varying voltage level of the battery. This was only realised by the team after the completion of the heats.

Overall, the outturn of the buggy was great considering the budget used was only £3.60 and it achieved the majority of the tasks that it was programmed for.

4.2 Total Cost and Marketability

The ESP project was a project that was made with the goal of producing a non-profit white line-following buggy for the purposes of serving as the prototype of future autonomous vehicles. Despite this, the marketability of the buggy is still quite high considering its performance during the Heats. However, Careful consideration would have to be made in regards to the costs of the design and manufacturing to bring the product to an acceptable profit margin in future production. The overall cost of the buggy constructed and tested by the team is about £230 if including all the free issue components during the project and two failed PCB boards due to incorrect printing. Approximately 200 hours were spent throughout the entire project to ensure the performance of the buggy, which includes stages of designing, prototyping, and testing. Based on the factors mentioned above, the selling price of the buggy designed by the team should be in the range of £270-280, which gives a considerable profit ratio of about 20% for each product. The target market for this product would be researchers of autonomous vehicles and educational organisations for educational purposes.

5. Analysis of Heats

5.1 Preparations for the Heats

The heats have been the goal of the embedded systems project and preparations were begun since the very start of the project. The buggy was designed specifically to overcome the obstacles present during the Heats and to achieve this with the least amount of time to outperform other competitors. This was done via the careful planning and development of the buggy along the full course of the project in various stages, following the standard procedures adopted by industrial professionals - that is designing, testing and improving. There were various technical demonstrations across the full period of the project; each technical demonstration served as benchmarks, testing the various components that form the racing system of the buggy. The first three technical demonstrations served as preparation for the heats.

The first technical demonstration (TD1) assessed the ability of the buggy's motor as well as the assembly of the buggy to ensure that the buggy is sturdy and has a pair of fully functional motors capable of moving and controlled via a PWM signal. TD1 allowed for the basic movement of the buggy to be created.

The second technical demonstration (TD2) assessed the capability of the sensor system of the buggy and the use of a Bluetooth signal to send a signal to the buggy externally to ensure that the buggy had a functional white-line sensing system to detect various scenarios that the buggy would go through during the Heats. TD2 created a sensing system for the buggy to detect changes to the white-line and provide inputs to the microcontroller.

The third technical demonstration (TD3) assessed the steering and control of the buggy to ensure that the buggy was capable of turning around corners and adjusting the speed of the wheel accordingly when faced with a slope.

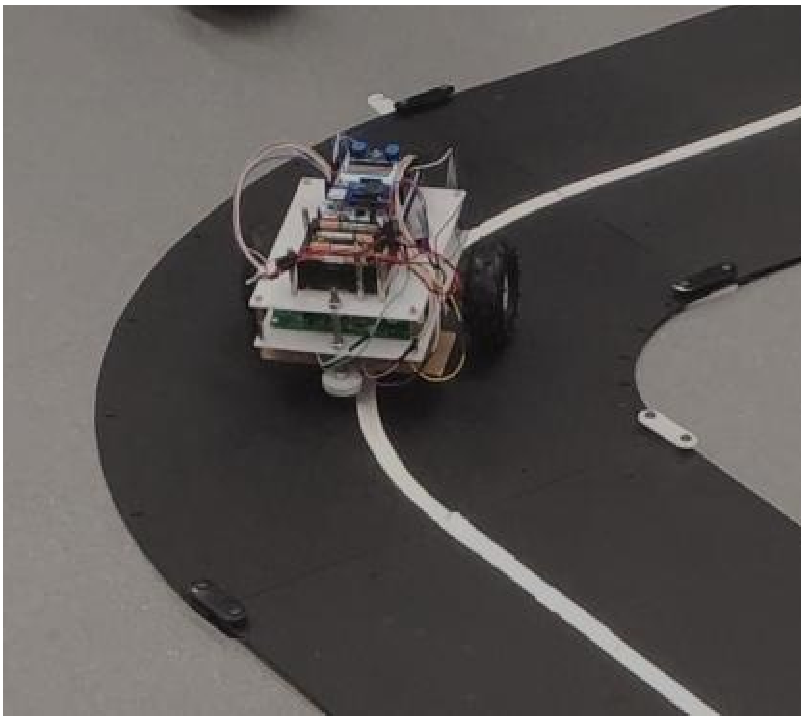

Figure 5.1.1: Test of buggy around left corner

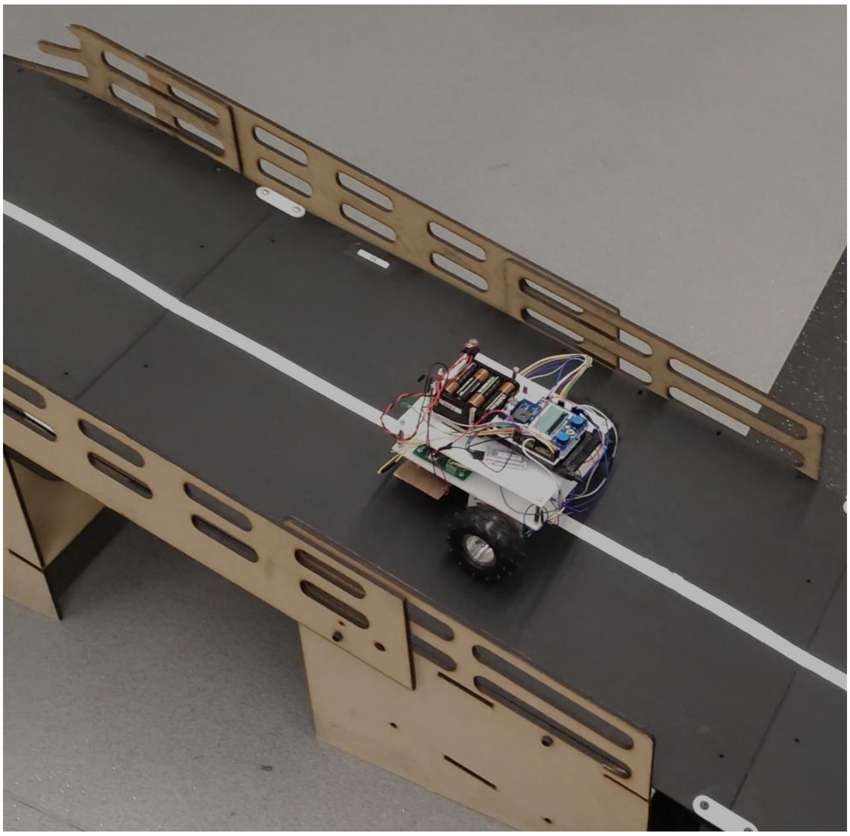

Figure 5.1.2: Test of buggy going uphill

The combinations of all the components developed at each technical demonstration forms the final racing system of the buggy which was then tested on an official piece of race track provided by the university as shown above in figure 5.1.1 and 5.1.2. In the case where the official race track was not available, the team made a race track resembling the official race track to further test the capability of the buggy with slightly different conditions in an attempt to improve the core performance of the buggy.

5.2 Summary of Buggy Performance During Heats

In the heats, the buggy managed to achieve most of the tasks, which included: successful navigation of a straight section of track, successful navigation of a left turn on the track, and successful navigation of a right turn on the track. Successful navigation of the pinch point, successful completion of uphill, successful completion of downhill and a successful controlled stop at end-point. The only task that was unable to be achieved by the buggy was the turnaround.

In the first run, the buggy managed to navigate through the straight, left, right and slope section of the track before arriving at the point of the turnaround, where the buggy's Bluetooth disconnected resulting in a straight crash into the barrier wall at the end. The disconnection of the Bluetooth was believed to be due to the limited range of the Bluetooth and the team member responsible for sending the Bluetooth stood too far away from the buggy.

In the second run, the team took note of the issue with the Bluetooth and decided that the member responsible for the Bluetooth control should follow the buggy as it navigates through the track. However, the buggy failed to make the right turn due to the voltage of the battery being too low, causing the speed of the buggy to drop significantly as compared to the first run. This resulted in the speed control algorithm of the buggy to take control, providing the buggy with a slight boost in voltage boosting the speed of the buggy, which caused conflict with the right turn speed setting, resulting in the buggy just running off track.

Due to uncontrollable factors in the first and second run, the buggy was unable to perform all the tasks that it is designed for, and the team opted for a third run, which prevented the buggy from getting timed. In the third run, the buggy managed to navigate through all the sections except for the turnaround. The turnaround was not achieved by the buggy and the buggy overturned and crashed into a wall. The Bluetooth signal was sent correctly and the buggy attempted to turn but failed, showing that this was due to the method of turning the buggy and not related to the Bluetooth. The turnaround was hard coded by the timing, meaning the buggy turned for a specified number of seconds which was set to 3 seconds. The turnaround of the buggy had been tested various times on the official track and a self-made track, however a perfect timing to turn the buggy around 180° was not found as this method of turning the buggy requires the voltage of the battery to remain constant, which is hard to achieve. The difference in voltage would cause the turn around speed of the buggy to vary, which causes the buggy to either overturn or under turn, thus the turning of the buggy was inconsistent and needed to be improved.

5.3 Critical Analysis of Buggy Performance During Heats

The performance of the buggy during the heats differed significantly between the individual tasks required to complete the track.

For simpler tasks such as the straight-line driving and corner turning, the code incorporated into the buggy was sufficient to propel the buggy around the straight line and corners extremely smoothly. The code incorporated provided the ability for the microcontroller to read the voltage values from each sensor and manipulate the left and right motor speed using a PID controller. This PID controller design proved to be effective in keeping the buggy as close to the white line as possible as during the heats, there was almost no deviation from the white line during the buggy's movement, indicating that the proportional, integral, and differential constants used in the PID controller were the correct values for the desired speed for the buggy. Although the smoothness of the buggy's movement around the straight line and corners was exceptional, the speed at which the buggy navigated these sections could have been improved. A main choice of improvement would be to increase the base duty cycle of each motor, to increase the normal speed of the buggy, and then tune the PID constants again to make sure the buggy still navigates each corner smoothly.

The pinch point section of the heats track was the same as the turns in functionality, therefore meaning it was also easily accomplished by the buggy for the same reason as the normal left and right turns.

The uphill and downhill sections of the heats track required speed control code for the buggy, which during the heats could be analysed as sufficient for the track but lacking in efficiency and response speed. During the heats the buggy managed to navigate the uphill and downhill sections of the track successfully, however, due to the design of the code incorporated, the buggy remained still at the start of the slope and then increased its speed to navigate the slope. This is due to the speed control code consisting of a speed bracket system, where the speed of the motors would be affected in three different discrete ways depending on the perceived speed of the buggy using the encoders. If the buggy was perceived by the microcontroller to be at a standstill or near-standstill, the microcontroller would add a specified increase to the duty cycle value of each motor, providing a high enough duty cycle for the buggy to climb the slope. If the buggy then reached a normal speed, the boost to the duty cycle of the motors would then be removed, therefore making the buggy move at a standard speed. This was also introduced for the downhill section in the form of a negative boost to the duty cycle, effectively causing the buggy to apply a reverse duty cycle to the motors when going down the slope. This resulted in the behaviour shown in the heats, in which the buggy temporarily stopped at the bottom of the slope, before increasing its duty cycle and then climbing the slope. When going down the slope the buggy would reduce its speed temporarily, speed up again due to the buggy changing back to a standard duty cycle, and then slow down again due to the negative duty cycle being applied again. The smoothness of this part of the heats track could be improved greatly by adding a proportional speed control, to provide a less drastic change in duty cycle depending on the difference between the buggy's measured speed and desired speed. This would reduce the response time of the speed control of the buggy and improve the accuracy of the buggy when navigating the track, as the sudden movements shown in the heats could cause errors in following the white line, due to the sporadic changes in speed (and potentially direction).

The turnaround point in the track was the one section the buggy could not complete correctly during the heats. The code designed for this part of the track was inferior to the other code used for the buggy as it relied on a set duty cycle and turn time in order to rotate the buggy 180°. This proved to be unreliable during the heats as the speed of the buggy during the turnaround point of the track varied depending on the energy level of the battery pack, as the turning time used during the heats was tested and performed correctly in previous testing with a different battery energy level. One of the successful aspects of this portion of the buggy's code design was that the activation of the turnaround point was relatively simple and easily tuneable, however during the heats the buggy's turning time was not able to be altered to the correct time for the heats track. An improvement for this aspect of the buggy would be to introduce a new set of code for the turnaround point, consisting of an identifier which would be activated once the sensors have seen the white line again, and indicate to the microcontroller that the buggy has found the white line again after starting to turn around. This would then cause the buggy to drive forwards again as normal, removing the need for a fixed turn time, greatly improving the reliability of the buggy.

The final part of the heats track was the controlled stop point. This section proved to be reliable and very successful, as the buggy stopped in a safe and controlled manner, using relatively simple code. The code used for this section of the track used the microcontroller and line sensors to check to see if no line sensors detected the white line, increasing a counter for every consecutive instance that the white line was not detected. After a short delay, the buggy would then reduce the duty cycle of both motors to 0.50 to stop the buggy immediately, then disable the wheels and turn on a blue LED to show the buggy has finished stopping. There were no possible improvements to this part of the buggy's behaviour.

5.4 Changes Made During Heats

There were no changes made to the code during the heats, as the first two runs were interrupted by factors that were outside the coding of the buggy. The turn around timing for the buggy was not changed as there were no chance to test it in the first two runs, and the team was unaware of its capability on the official race track. During the second run of the TD4 track, the battery level was shown to be inefficient as the buggy activated its speed control boost when going around the right bend as the wheels were moving too slowly. The wheels were moving too slowly due to a decreased battery level which reduced the maximum speed of the buggy, causing the microcontroller to read a low speed from one of the wheel's encoders and anticipated the slope when the slope hadn't been reached yet. To counteract this, the batteries were charged between the second and third run which allowed it to maintain a sufficient buggy speed for the third run.

5.5 Most Successful and Worst Feature of the Buggy

Through analysis of the heats, the best feature of the buggy was clearly shown to be its PID controller as the buggy did not deviate from the line during the heats, with this accurate line following being directly controlled by the PID controller. This means the proportional, integral, and differential constants were perfectly implemented for the speed the buggy was travelling at, making the PID controller the most successful part of the buggy. This was further proven as the pinch point requirement of the heats track tested the accuracy of the PID controller by providing a section of the track with a smaller minimum track width [1], forcing the buggy to maintain a tightly constrained control over its deviation from the white line.

The worst feature of the buggy was shown to be its turnaround point. This was shown during the third run of TD4 where the buggy overturned and could not drive back down the track in one go. This is due to the limitations of the code used for the turnaround point, as described in part 5.3 of this report.

6. Executive Summary

To summarise, the buggy's mechanical components were made using a set of motors along with a set of the chosen gearbox 2, fitted on a two-level chassis designed to fit all necessary components. The electronic components consisted of six TCRT5000 sensors arranged in a straight line to form the sensor system. This was then connected via the microcontroller to the main system consisting of the motor driver board, which provided power to the microcontroller, motors and encoders from the batteries. The system also used encoders that formed the speed sensing system for the buggy, and a bluetooth module which allowed for external commands such as a turnaround to be programmed on the buggy. The control components consisted of the PID, speed control and turnaround algorithm to form the behaviour of the buggy. The software component was a piece of code that integrated everything together.

The team organisation and planning contributed greatly to the successful completion of the embedded systems project. By following the Gantt chart, team members were able to differentiate all the necessary tasks to complete the group deliverables. The team roles were divided between software and hardware to ensure a smooth workflow and equal participation. The difficulties that arose with the buggy's coding and assembly were quickly detected and members were able to collaborate and find efficient solutions to guarantee a positive outcome in the technical demonstrations. Teamwork could have been improved during the project's runtime by assigning a project leader early on to delegate the tasks and improve decision making by guiding team members to engage fully with the module.

The budget provided for the team was more than sufficient as there were not any significant purchases made by the team except for five additional TCRT5000 sensors for a total cost of £3.60, bringing the total budget down to £36.40. Considering the budget spent on the buggy, the outturn of the buggy's performance was beyond expectation, though it could be further improved, had the team fully utilised the budget at hand. The estimated price of the buggy would be around the range of £270 - £280, which would give a profit range of £40 - £50 per unit sold.

The preparation of the heats had been sufficient and the buggy passed the various benchmarks set along the course of the project. The performance of the buggy during the heats was great. The buggy has managed to achieve most of the tasks it was designed for despite the various unexpected events that occurred and hindered its performance. There were no huge changes made for the buggy in the heats other than the recharging of the batteries to restore the energy level of the batteries to the level the buggy is designed to operate on. The greatest feature of the buggy was its PID algorithm which was finely-tuned for a great period of time, allowing the buggy to achieve smooth turns around the corners. The worst feature of the buggy was the turnaround algorithm, as its performance was not consistent and was highly dependent on the energy level of the batteries and timing.

7. References

[1] The University of Manchester, Embedded Systems Project Procedures Handbook Version 2022.2, February 2023.

[2] The University of Manchester, Embedded Systems Project Technical Handbook, 2022.

[3] A. Slater, F. Kougionas, A. Kamaruddin, K. T. K. Min and T. Wang, "Design Report 1", Manchester, 2022.

[4] A. Slater, F. Kougionas, A. Kamaruddin, K. T. K. Min and T. Wang, "Design Report 2", Manchester, 2022.

[5] A. Slater, F. Kougionas, A. Kamaruddin, K. T. K. Min and T. Wang, "Proposal Document", Manchester, 2023.