

FACULTY OF MECHANICAL AND AUTOMOTIVE ENGINEERING TECHNOLOGY

TEAM ORIENTED PROJECT STUDIES (TOPS)

BHA 4704

PROPOSAL PROJECT TITLE:

**DEVELOPMENT OF TEST PLATFORM FOR VRU SOFT
TARGET**

NO.	NAME	MATRIC ID
1.	MUHAMMAD MUJAHID BIN MANSOR	HB19009
2.	HAZIQ BIN ABDUL RAZAK	HB18012
3.	HASIF FARHAN BIN HANISOFIAN	HB19025
4.	DESMOND LING ZE YEW	HB19024

PROJECT SUPERVISOR: DR. MOHD NADZERI BIN OMAR

HsKA SUPERVISOR: PROF. DR. –ING. MAURICE KETTNER

Table of Contents

ABSTRACT	4
1.0 INTRODUCTION	4
1.1 Problem Statement	4
1.2 Objective	4
1.3 Project Scope	5
2.0 LITERATURE REVIEW	6
2.1 Advanced Driver Assistance System (ADAS)	6
2.2 Vulnerable Road Users (VRU) Soft Target	8
2.3 Platform for VRU Soft Target	10
2.4 Criteria for the Platform for VRU Soft Target	11
2.5 Existing Product	12
2.5.1 AB Dynamics Launchpad	13
2.5.2 GPS Guided Robot	16
2.5.3 Agricultural Robots for Organic Farming	18
2.4 Self-driving Bicycle	20
3.0 METHODOLOGY	22
3.1 Design Conceptualization	22
3.1.1 Design Concept #1	22
3.1.2 Design Concept #2	23
3.1.3 Design Concept #3	23
3.1.4 Design Concept #4	24
3.1.5 Design Concept #5	25
3.1.6 Design Concept #6	25
3.1.7 Design Concept #7	26

3.1.8 Controller Design	27
3.2 Design Evaluation and Selection	28
3.2.1 Evaluation Table	28
3.2.2 Finalized Design	31
3.3 Materials Selection.....	32
3.4 Machining	32
3.4.1 Jigsaw Machine	
32	
3.4.2 3D Printer	
33	
3.5 Expected Bill of Materials and Cost Estimation	Error! Bookmark not defined.
4.0 EXPERIMENTAL PROCEDURE	64
4.1 PROCEDURE.....	65
4.1.1 Development and Testing	65
4.1.2 Assembly of the platform.....	65
4.1.3 Assembly of the remote controller	66
4.2 EXPECTED RESULTS	66
5.0 CONCLUSION	67
References	
68	
APPENDIX	
71	
GANTT CHART	
71	

ABSTRACT

This project is aimed at developing a VRU soft target platform that is portable and robust, easy to operate, low maintenance, and long hours of use for the usage by students and lecturers for learning purposes and experiments. The main objective of this project is to design and fabricate a movement and steering function for test platform for VRU soft target controlled wirelessly. The design will utilize two microcontrollers, namely the ESC for the movement and Arduino for the steering system. The control interface with the Arduino is developed by using a controller to instruct movement controls to the vehicle. The experimental results showed that the accuracy rate of the robot is 83.33%.

1.0 INTRODUCTION

Accidents involving road users cannot be overlooked and according to [1], National Safety Council estimate more than 46 000 people were killed in 2022 from a motor vehicle crashes, which is an increase of 22 percent of mileage death rate compared to 2019 during pre-pandemic. Cars nowadays are mostly equipped with Advance Driver Assistance System (ADAS) from luxury car to affordable cars in the market as a way to prevent road collision. Hence, for testing the built-in ADAS in the car, proper tool is required. Thus, this project is being proposed in developing a VRU soft target platform starting from the movement and steering for the test platform according to the behavior of the vulnerable road user such as pedestrians, cyclists and motorists. In this project, we will be covering the maneuverability of the said test platform by using Arduino, controller, sensors as well as motors for the movements. The study could be used as future references for the application of the test platform for VRU soft target to be used in industry and to improve our knowledge about the current development of VRU soft target test platform.

1.1 Problem Statement

ADAS system is now a common thing that most of car manufacturers applied to their cars. Hence, the need of robust and suitable test platform for the VRU soft target which includes the maneuverability and durability of the steering system of the project becomes crucial for this project. One of the challenges is that it requires wireless controllability as well as preset route for the VRU soft target test platform to perform such test at ease. The platform should withstand high impact from a load of a

vehicle as well as traction to keep precision the route of the test platform during testing.

1.2 Objective

1. Design and fabricate a movement and steering function for test platform for VRU soft target controlled wirelessly.
2. Develop the coding for the control interface with the Arduino by using controller to instruct movement controls to the vehicle.
3. Create the mechanical and electrical mechanism of the steering of the test platform.
4. Calculate and analyze suitable degree of turning angle values to operate the vehicle.
5. Perform field tests for functioning abilities.

1.3 Project Scope

The project is to conduct an experiment of applying the movement and steering system to a VRU soft target test platform and designing a test rig or test bench for the ADAS vehicle. An experiment will be conducted to conclude the overall results.

2.0 LITERATURE REVIEW

CHAPTER 2: LITERATURE REVIEW

2.1 Advanced Driver Assistance System (ADAS)

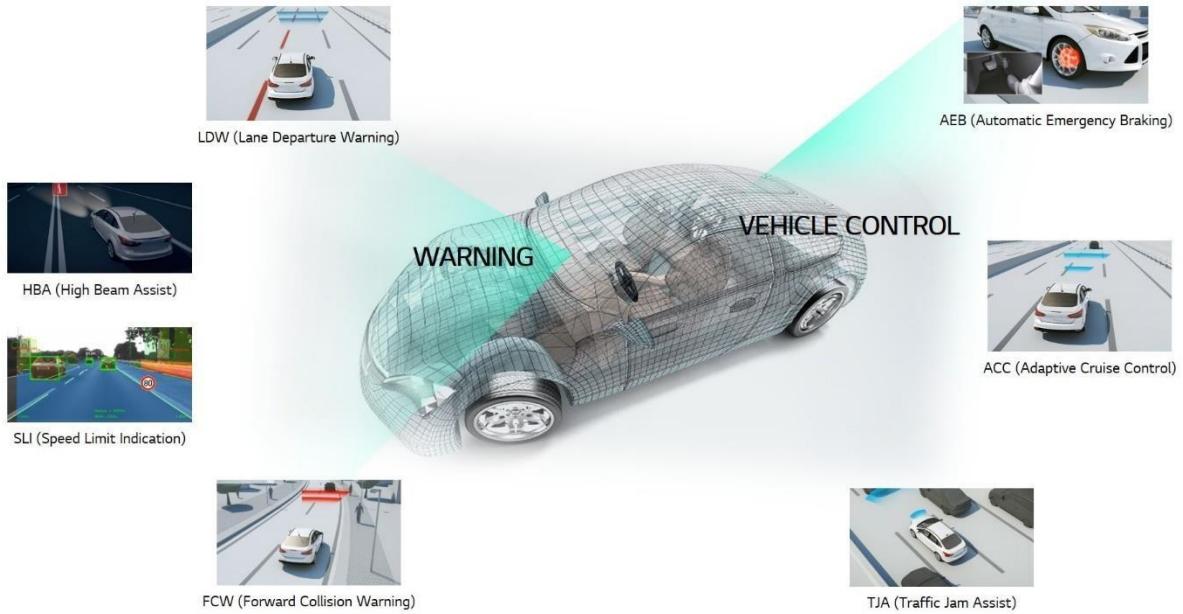


Figure 2.1: Advanced Driver Assistance System (ADAS).

(Source: <https://www.lg.com/global/mobility/press-release/lg-to-supply-next-generationadas-cameras-to-premium-german-automaker>)

Figure 1.1 shows the Advanced Driver Assistance System (ADAS) in a modern passenger vehicle nowadays. As shown in the figure above, the ADAS gives the features of Lane Departure Warning (LDW), High Beam Assist (HBA), Speed Limit Indication (SLI), Forward Collision Warning (FCW), Automatic Emergency Control (AEB), Adaptive Cruise Control (ACC), and Traffic Jam Assist (TJA). Advanced driver assistance systems (ADAS), which are intended to reduce or prevent crashes in passenger vehicles, are changing how we drive. Given the over 37,000 traffic fatalities recorded in the US in recent years, there is mounting evidence that ADAS are successful in lowering crash rates [1]. However, in order for society to fully benefit from ADAS's improvements to highway safety, drivers must not only embrace their vehicles' ADAS features but also comprehend their advantages and disadvantages in order to properly manage them. [2], [3].

Vehicle crashes can be reduced using ADAS. Forward Collision Warnings (FCW) on vehicles resulted in fewer and less serious collisions [4], yet the best results were seen in vehicles having both FCW and Automatic Emergency Braking (AEB) systems [5]. When combined with AEB, FCW reduced rear-end struck crash rates in passenger cars by 50%, compared to 27% for FCW alone [6]. AEB also decreased crashes involving large trucks by more than 40% [7]. Blind spot monitoring [8] and cross-traffic alerts [9] also decreased collisions. This evidence demonstrates that ADAS technologies have the potential to significantly improve public safety.

Due to ADAS's limitations, automation must be under supervision. However, in subsequent NTSB-investigated crashes, the automation was found to have failed when the driver was not supervising it. Tractor-trailers crossed the path of the car in two of the tragic collisions, but neither the Level 2 ADAS nor the driver used the brakes [10]. These crash investigations are in line with a meta-analysis's conclusion that greater automation is linked to worse driver recovery from a system breakdown [11]. According to a recent study, owners of cars with adaptive cruise control (ACC) were no more knowledgeable about the system's limitations than non-owners [12]. Inadequate conceptual understanding of ADAS may deter drivers from investing in ADAS-equipped cars, deter drivers from using ADAS in their cars, and deter drivers from supervising ADAS in their cars.

2.2 Vulnerable Road Users (VRU) Soft Target



Figure 2.2: A VRU soft target

(Source: <https://www.autonomousvehicleinternational.com/news/adas/vehicle-testing.html>)

Figure 2.2 shows the computer-controlled carrier for vulnerable road user (VRU) soft targets which was developed by AB Dynamics, designed for use in ADAS and autonomous vehicle test scenarios. As a category of vulnerable road users (VRUs), pedestrians, cyclists, and operators of motorised two-wheelers are among those who are most at risk on European roads, contributing to 46% of traffic fatalities and 51% of serious injuries from 2009 to 2015. [13]

Statistics from the World Health Organization (WHO) show that 1.35 million people die globally each year as a result of traffic accidents. VRU deaths make up 54% of all traffic fatalities, with pedestrians, cyclists, and motorised 2-3 wheelers making up 23%, 3%, and 28% of those fatalities, respectively [14]. For instance, from 2009 to 2015, in Germany, motorcycle riders accounted for 17% of all fatalities, followed by pedestrians (15%), cyclists (11%), and moped riders (2%) [15]. Based on data from traffic accidents in 2005, pedestrians make up 37% of fatalities in Japan, followed by cyclists (14%), motorcyclists (11%), and moped riders (6%) [16]. According to the China In-Depth Accident Study (CIDAS), pedestrians and twowheelers account for roughly 25% and 40%, respectively, of all traffic deaths in China [17]. According to the above statistics, in the majority of motorized regions, VRU accounts for more than half of all fatal traffic accidents. Because of this, the safety of VRU has received a lot of attention globally.

For this instance, Autonomous vehicles (AVs), safety systems, and advanced driver assistance systems (ADAS) have the potential to make these people's interactions with traffic much safer. To successfully complete the testing procedures, the scenario complexity and heterogeneity of the VRUs must be modelled using corresponding test tools and VRU dummies, or targets. Numerous software and hardware test cycles are undertaken during the development of these systems, and each cycle necessitates validation by real-world test scenarios on proving grounds. Before the cars can move on with serial manufacturing, only the testing of every component in an integrated system can ensure a trustworthy judgement on the functionality. The challenge for the test procedures is to realistically reproduce traffic situations, especially critical boundary cases, and thus to test the intervention, prevention or mitigation of critical collisions between vehicle and VRU in a reproducible way under real conditions with VRU soft targets on the test site. [13]

2.3 Platform for VRU Soft Target



Figure 2.3: Platform for VRU soft targets.

(Source: <https://www.youtube.com/watch?v=x7-SS1LxjPw>)

Figure 2.3 shows the various ADAS scenarios performed using AB Dynamics' compact powered platform LaunchPad with adult, child and cyclist VRU soft target. As ADAS and autonomous vehicle technologies continue to advance, so too does the need for rigorous testing to ensure that vehicles are safe for public roads. That's why a highly maneuverable ADAS target platform has to be designed specifically to replicate challenging urban test scenarios. In order to carry VRU (Vulnerable Road User) targets for ADAS research and testing, a platform has to be designed and fabricated. For example, as shown in Figure 2.3, LaunchPad is a pilotable platform developed by AB Dynamics. It can go at a top speed of 50 kph while carrying a target, making it appropriate for both pedestrians and dummies riding bicycles, mopeds, and scooters. Because of its durability, the chassis can withstand being run over by a car. LaunchPad also enables the next generation of ADAS tests by allowing the design of pathways and speed control profiles. To allow exact choreography with the target vehicle, LaunchPad uses Synchro and path following technologies [18].

2.4 Criteria for the Platform for VRU Soft Target

For our project, we are required to design and fabricate a platform for VRU soft target.

The design of the platform should follow some of the criteria, as stated below:

a. Movement

- The platform should have the ability to move forward and backward.
- The motor should have high torque to overcome both the weight of the platform and the soft target.
- The platform should be programmable and can follow a preset route.
- The platform should be able to maintain a constant speed of 5-10km/h.
- The platform should be installed with a functional braking system (exp. Regenerative braking).

b. Steering System

- The platform should be able to turn left and right.
- The complexity of the steering system for the platform should be minimized for time and cost saving.
- The wheels of the platform should have a high traction force to grip the road surface during cornering.
- The platform should be able to move and turn easily even though loaded with the heavy soft target or in different road conditions.
- The durability of the steering system should be maximized.

c. Control System

- The platform should be able to control manually by using a distributed controller.
- The design of the control system should be less complex and user-friendly.
- The platform should have a wireless and wide range of connectivity due to the outdoor testing.

d. Cost

- The materials and fabrication costs should be lower than RM500.

- The fabrication of the platform should be simplified to reduce the cost.
 - The components used in the system should be minimized to save time and cost.
- e. Assembly
- The maximum height of the overall assembly should be less than 85mm.

2.5 Existing Product

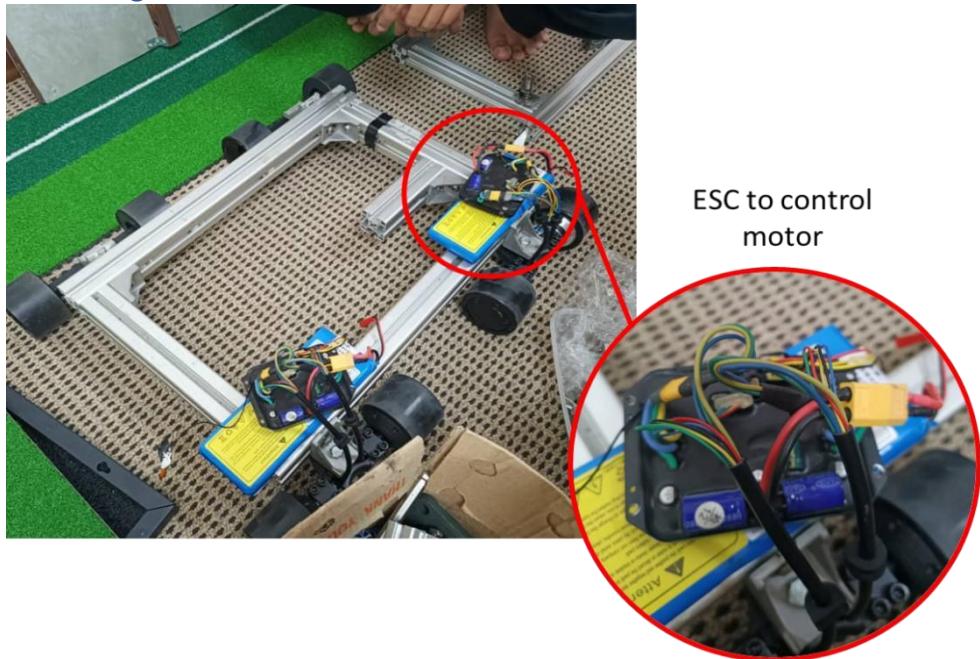


Figure 2.4: The current design of the platform.

The current design of the platform, as seen in Figure 2.4, utilizes 2 Electronic Speed Control (ESC) as the brain of the system to accelerate. Since the motor can only move in one direction, 2 motors are being used with one to accelerate forward while the other to reverse backward. The system also utilizes a differential steering system to turn left or right, which came to the need for us to improve the maneuverability and reduce the traction force during turning for the system. Therefore, our main concern for this project is to redesign the platform that is given to us, in order to meet the criteria that are mentioned in Chapter 2.4. A few currently existent online-only products are referred so that we can have more ideas to redesign the platform in terms of its steering system, control system, etc.

2.5.1 AB Dynamics Launchpad



Figure 2.5: The AB Dynamics company.

(Source: <https://www.abdplc.com/our-business/>)

AB Dynamics, as shown in Figure 2.5, is a vehicle engineering firm that was founded in 1982, has progressively expanded to become one of the most reputable manufacturers of automotive test equipment. Today, the top 25 automakers worldwide, each of the seven Euro NCAP testing facilities, and various government test agencies are among their clientele.

With effective data and protocol integration from virtual to physical to real-world, applications range from extremely efficient durability testing to precision control for crucial new fields of technology development like active safety and autonomous driving [19]. Until today, they are one of the top companies that produce and sell high quality VRU soft target platform, namely LaunchPad. The LaunchPad range of powered platforms has been created to carry pedestrian, cycling, moped and motorcycle dummies as Vulnerable Road User (VRU) targets for ADAS and autonomous vehicle (AV) research and testing. The testing is accurate and repeatable thanks to the range's durability. The LaunchPad can take intricate trajectories that are not limited to straight lines, unlike belt-driven platforms [20]. There is a few LaunchPad designed to meet customers' specific testing requirements, ranging from low-speed pedestrian testing through to highway speed motorcycle testing. The LaunchPad range includes the LaunchPad Spin, the LaunchPad 50 and 60, and also the LaunchPad 80.



Figure 2.6: LaunchPad Spin

(Source: <https://www.abdynamics.com/en/products/track-testing/adas-targets/launch-pad>)

The third-generation platform and newest addition to the LaunchPad family, the LaunchPad Spin, as shown in Figure 2.6, gives VRU testing new levels of maneuverability. Due to its turn-on-the-spot, single-wheel steering, which was designed to emulate difficult urban conditions, the testing can accurately mimic the quick changes in direction that are typical of real-world pedestrian behaviour. Additionally, because of its top speed of 30 km/h, it can test situations involving e-scooters and bicycles [20].



Figure 2.7: LaunchPad 50 and 60.

(Source: <https://www.abdynamics.com/en/products/track-testing/adas-targets/launch-pad>)

For the development and testing of ADAS, the LaunchPad 50 and LaunchPad 60, as shown in Figure 2.7, are faster platforms made to transport targets for Vulnerable Road Users (VRU). The LaunchPad may be utilized with pedestrian, bicycle, moped and motorcycle dummies with top speeds of 50 and 60 km/h, respectively, and has precise speed control and full path-following functionality [20].



Figure 2.8: LaunchPad 80.

(Source: <https://www.abdynamics.com/en/products/track-testing/adas-targets/launch-pad>)

Even under the most difficult circumstances, realistic highway speed ADAS testing with motorcycle targets is made feasible by the LaunchPad 80, as illustrated in Figure 2.8, is currently the fastest VRU test platform that developed by the AB Dynamics company. The LaunchPad 80 offers the speed and quick deceleration required to conduct realistic motorcycle test situations, with test speeds of 80 km/h and 0.6 g. It is the first solution to fully satisfy the 2023 Euro NCAP AEB/LSS protocols' speed requirements [20].

2.5.2 GPS Guided Robot

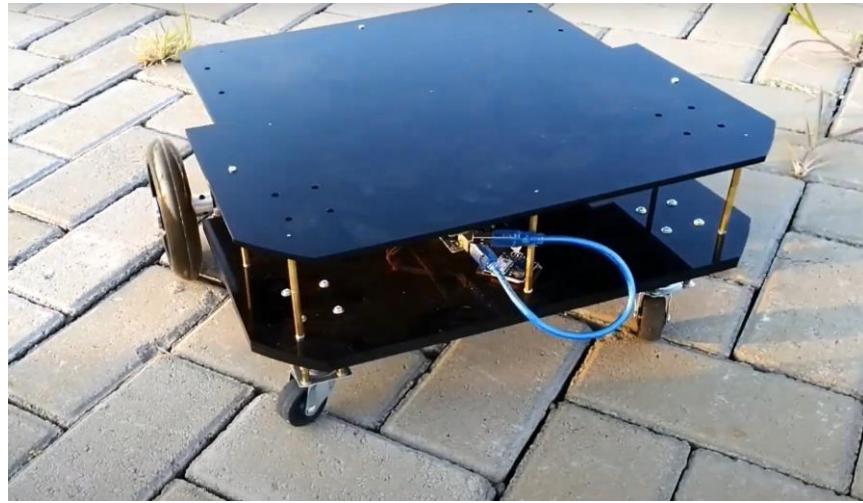


Figure 2.9: A GPS guided robot with differential drive system and caster wheels.

(Source: https://www.youtube.com/watch?v=GkXH8ZeeIKY&t=48s&ab_channel=Firebitlab)

As shown in Figure 2.9, it is a GPS guided robot that was developed by Richard Kurniawan, a student from the Faculty of Information Technology, Ciputra University. As shown in the paper ‘Design and Implementation prototype of goods carrier robot using microcontroller’, he actually developed a robot prototype that aims to help human to transport the goods. With the development of this technology, it is able to help industry to save a lot of human labor costs. In order to make the automatic goods transport robot run towards the coordinate point, this robot is equipped with a Bluetooth module, GPS module, and IMU sensor. This robot is using the differential drive system in which its movement is based on two separately driven wheels placed on either side of the robot body. It can thus change its direction by varying the relative rate of rotation of its wheels and hence does not require an additional steering motion. Two passive caster wheels are attached to the bottom of the robot for balancing purposes. Throughout ten experiments performed on the robot, the accuracy rate of the robot is 83.33%. The failure caused during the experiment is estimated due to the satellite signal is blocked by the building or trees around the location, in which the signal strength is weakened and caused the level of accuracy of the GPS to decrease [21].

There are several advantages and disadvantages for this system as explained below:

a. Advantages

- Simple and robust: As mentioned in ‘Robot Modeling and Control’, by Spong, M.W., Hutchinson, S. and Vidyasagar, M. (2006), differential drive systems have a simple mechanical design and are relatively robust. As in Figure 2.9, there is no need to modify a lot for the platform that is given to us as a separate servo motor for cornering is not needed. This increases the simplicity of the project [22].
- Low power consumption: Differential drive systems typically consume less power than other drive systems, as it does not require a separate servo motor or actuator for cornering. This could be particularly advantageous for our project.
- High maneuverability: Differential drive systems allow for precise control of the robot's movement and turning radius. This makes them well-suited for applications that require high maneuverability, such as indoor navigation or search and rescue. [22]
- Reduced cost: In ‘The Vector Field Histogram - Fast Obstacle Avoidance for Mobile Robots’, IEEE Journal on Robotics and Automation, 7(3), pp. 278-288, mentioned about due to the simple mechanical design, differential drive systems are often less expensive than other types of drive systems [23].

b. Disadvantages

- Limited stability: ‘Introduction to Robotics: Mechanics and Control’ by Craig, J.J, stated that differential drive systems can be less stable than other drive systems, particularly when operating on uneven terrain or at high speeds [24].
- Skid steering: Differential drive systems rely on skid steering, which can cause damage to some surfaces and result in increased wear and tear on the robot's wheels, especially the wheels that we used in our project which will frequently run outside above the rough road surface [22].
- Complex calculation: Differential drive systems can involve some calculations, particularly when it comes to control and navigation. For example, calculating the desired wheel speeds or velocities based on a desired trajectory or movement command can require some mathematical computations. However, with the availability of modern microcontrollers and programming languages, these

calculations can typically be implemented in software and performed quickly and efficiently. But still, it will take a lot of time to be carried out [25].

2.5.3 Agricultural Robots for Organic Farming

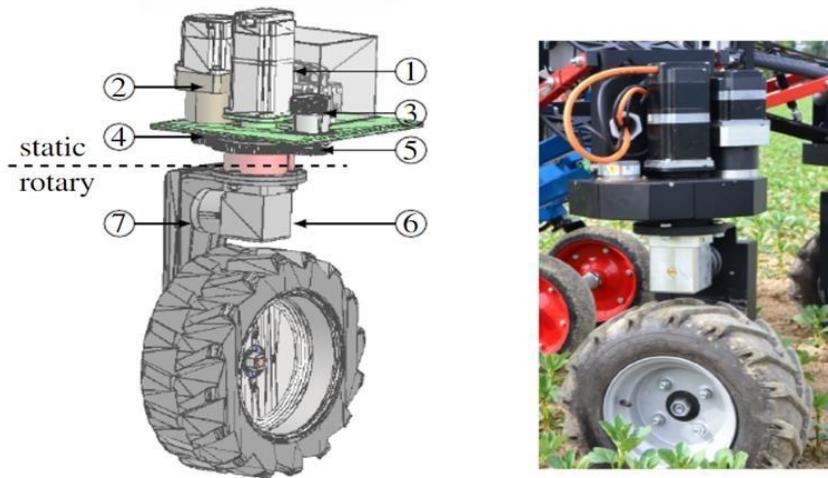


Figure 2.10: Agricultural Robots for Organic Farming

(Source: ‘Towards Agricultural Robotics for Organic Farming’, by Georg Halmetschlager, Johann Prankl, and Markus Vincze)

Figure 2.10 above shows the steering system for agricultural robot that used in organic farming which developed by Georg Halmetschlager, Johann Prankl, and Markus Vincze. Based on their paper ‘Towards Agricultural Robotics for Organic Farming’, they proposed a scalable and modular agricultural robotic concept that advances farming to the next higher technology level, due to the fact that the field of organic farming is still characterized by multiple manual tasks that include heavy labor. Therefore, they provided a low-cost and flexible design in order to realize different autonomous applications, specialized for light weight agricultural work. They utilized a local navigation system based on a self-parameterizing crop row detection, that enables local adaptable, and GPS-independent navigation. Then, the hardware and software of the designed system should be able to handle rough terrain, offer a high maneuverability, and is adaptable to different row-structures [26].

A robot needs at least three degrees of freedom (DoF) to move, as every car-like vehicle does. The classic kinematic realization of service robots are differential drives. But they decided to implement a n-wheeled steering to combine tractive power, maneuverability, and scalability of the robot. Therefore, they proposed a kinematic encapsulated powertrain that can be equipped with or without a motor for the steering or tractive power. The wheel will be a free running wheel without any motor, and equipped with a single motor for pure tractive power, or as fully powered, independent steerable wheel [26]. There are several advantages and disadvantages for this system as explained below:

a. Advantages

- The system costs less when compared to an electric steering system. Repairs can also be done using a basic knowledge of servo, reducing any costs related to repairs.
- Ease of repairs: The system is not complex and as mentioned above, the repairs can be done easily. Furthermore, no periodic maintenance is required [27].
- Ease of programming: Since the system is widely used including in RC steering system, the coding of the system has various sources and references to be studied when developing the necessary function of the system.

b. Disadvantages

- Low maneuverability: The steering mechanism has difficulty turning by direct connection of servo due to the design of the wheel used. This, however, can be overcome by placing bearing at the axle.
- Lower range of connectivity: The system required additional components to enhance the transmission or using totally different components for the connection such as Wi-Fi and Bluetooth.
- Less power: Although a servo motor is at high efficiency at light loads, it simply does not have enough power to maneuver the high load system of the VRU soft target platform. A higher torque servo motor may be required.

2.5.4 Self-driving Bicycle



Figure 2.11: The steering system of a self-driving bicycle.

(Source: <https://www.designboom.com/technology/self-driving-bicycle-huawei-engineers-operate-unmanned-06-14-2021/>)

Figure 2.11 shows a self-driving bicycle that can operate without any user, developed by Zhihui Jun, one of Huawei's engineers. There are sensors and cameras that allow it to be autonomous with the help of AI technology. The steering system is a simple belt/wire driven by a servo motor that helps it to turn the handles. This can be integrated into our VRU soft target platform as one of the methods of steering. The servo motor used is a 60kg or 6Nm torque motor (Figure 5.2) that is strong enough to steer the wheels [28].

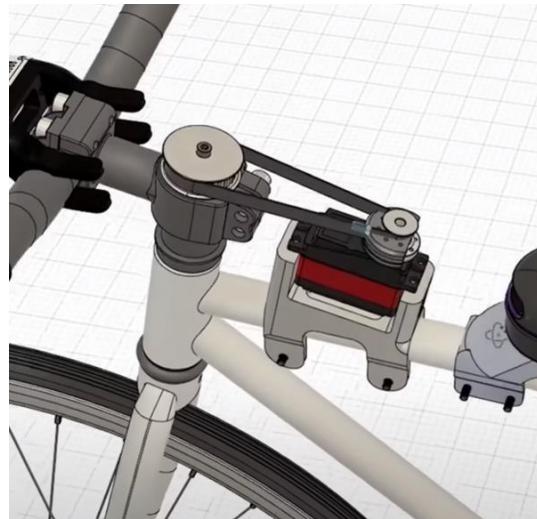


Figure 2.12: Design of the steering in CAD.

Based on the steering system of the self-driving bicycle as shown in Figure 2.12, it can be clearly seen that a timing belt acts as a connection between a servo motor and the shaft of the wheels. The wheel then can be turned by just controlling the rotation of the servo motor.

There are several advantages and disadvantages for this system as explained below:

a. Advantages

- Efficient: Low power loss.
- Low maintenance: No lubrication required.
- Cost effective: Belt made of rubber is cheap to replace.
- Ease of operation: Simple system.
- Smooth power transmission
- Light weight: Rubber is light.

b. Disadvantages

- Slip and creep may occur: Rubber belt may slip at higher torque.
- Belt wear: Belt may crack after long use.
- Deformation: Elongation of belt after continuous use.
- Lower operating temperature: The belt has lower operating temperatures

3.0 METHODOLOGY

3.1 Design Conceptualization

To improve the current design of the platform, we have gone through much research on the existing product as reference for our improvement as discussed in Chapter 2.5. The design also prioritized the criteria that we have set to be focused on for the project. Hence, there are 7 ideas on the design concept with different innovations and benefits that we will discuss below.

3.1.1 Design Concept #1

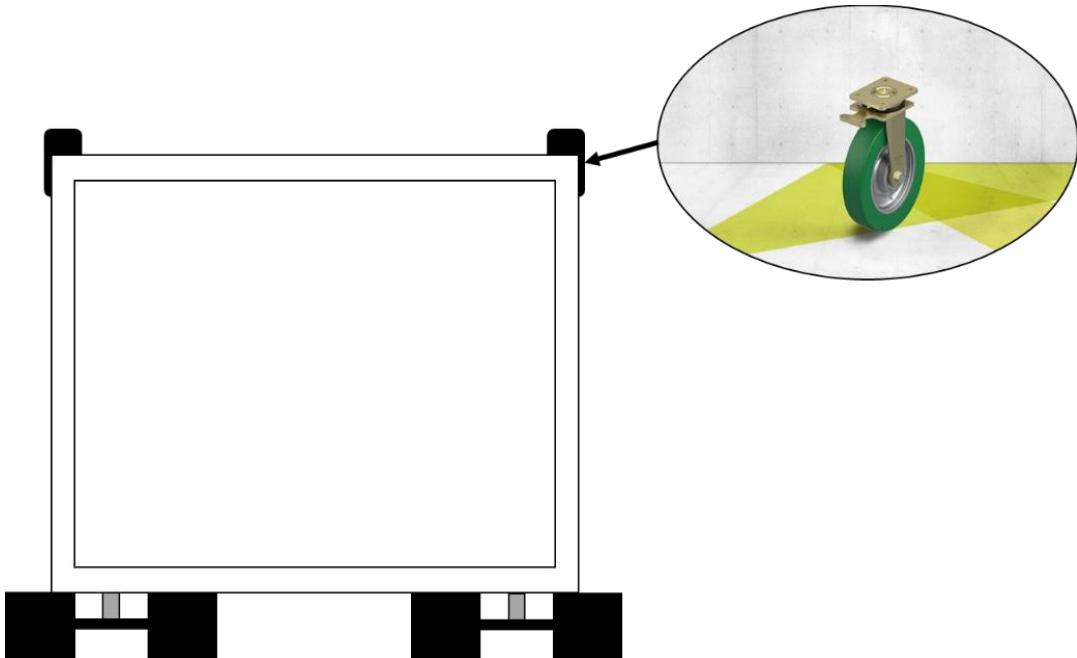


Figure 3.1: The sketching of Design Concept #1.

The design concept maintains the current concept in the of the steering system. The only difference is to change the front wheel with free moving wheels, also known as caster wheels, to ease the turning of the platform as shown in Figure 3.1. With this system, only a little change is required on the current platform and thus we will be able to reduce the fabrication cost of the project. The design also does not require any additional microcontroller or electrical system. However, it will involve a complex calculation to get the precise speed of the wheels' motor to

turn the platform into a desired angle due to the differential steering system, which manipulates the speed of each side of the wheels to turn left or right.

3.1.2 Design Concept #2

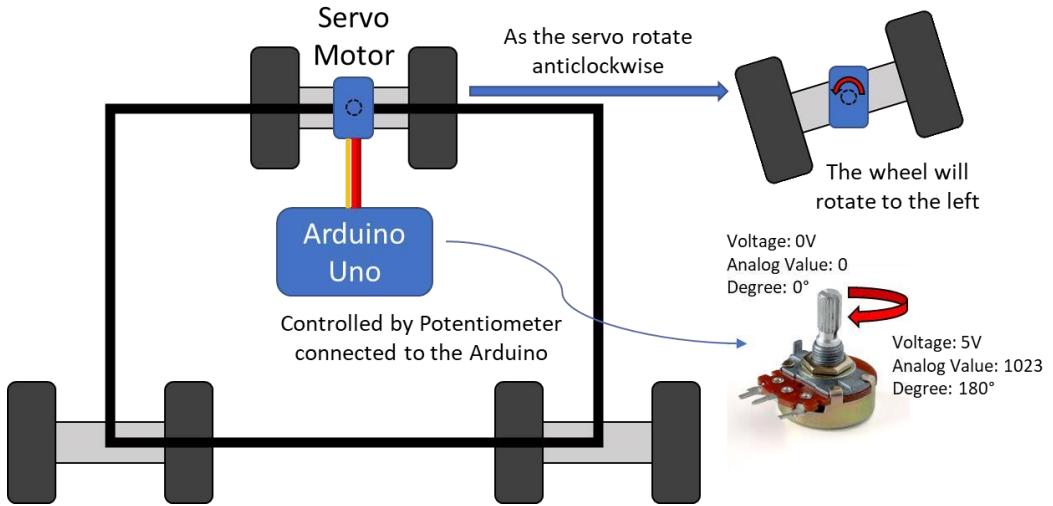


Figure 3.2: The sketching of Design Concept #2.

This design added a servo motor to control the steering of the front wheel while connected directly as shown in Figure 3.2. A bearing will also be added to the axle of the wheel to ease the rotational movement and by using the RF Transceiver (nRF24L01+), which can be connected via radio signal transmission up to 100m range in an open space and can be increased to 700m with addition of amplified antenna, a wireless connection is able to be established with an Arduino Uno used as the brain of the.

The design can be further improved by using caster wheels, which used the same concept as Chapter 2.5.3 Agricultural Robots for Organic Farming. This will increase its maneuverability while maintaining a reliable servo-based system. One of the downside of the system is that the height may be exceeding the allowed standard for the ADAS testing.

3.1.3 Design Concept #3

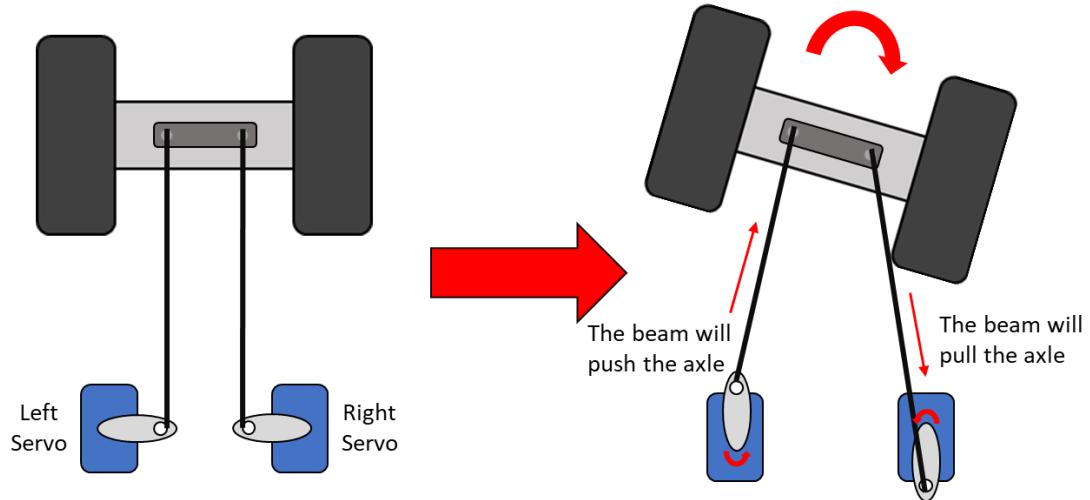


Figure 3.3: The sketching of Design Concept #3.

This concept utilizes 2 servo motors which are attached to the axle of the wheel via linkage bars or beams. Referring to Figure 3.3, the left servo motor's arm is facing to the right servo and vice versa for the right servo motor's arm for the initial position. When the servo motors rotate 90° anticlockwise, the beam on the left servo will push the axle while the right servo's beam pulls it, making the wheel turn to the right.

The basis of the concept needs to be improvised in a way that it can be used more practically, especially on the design of the linkage bars which show the complexity of the system. Furthermore, the robustness of the system may be concerning and unable to withstand the load from the impact of a car.

3.1.4 Design Concept #4

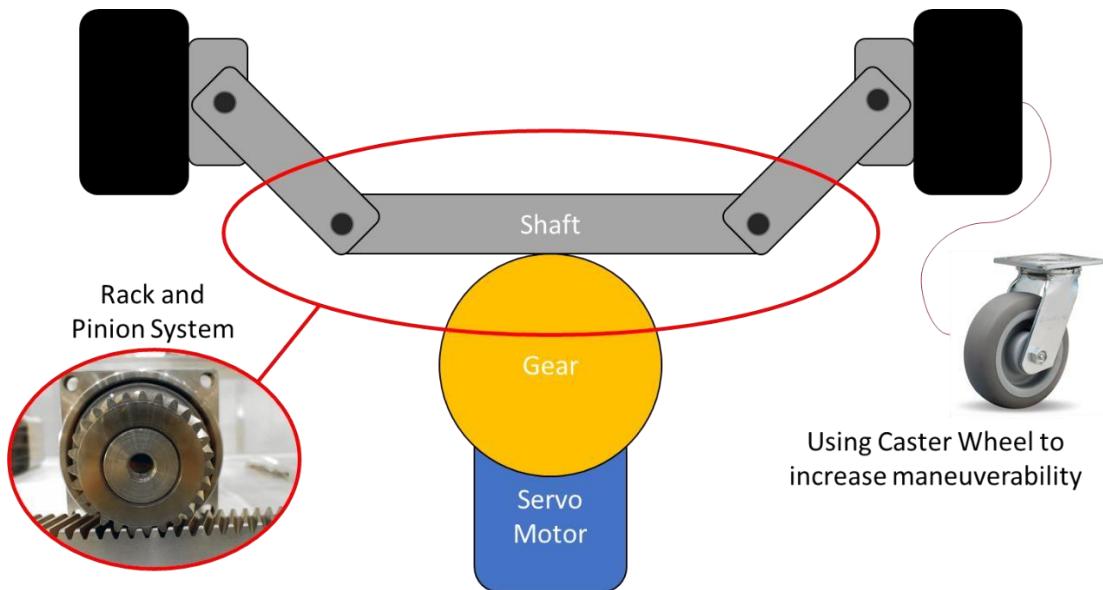


Figure 3.4: The sketching of Design Concept #4.

Another design that utilizes the servo motor as the steering system is by using gears for the connection with the wheels. This allows more precise turning compared to the linkage bars connection as the gears can be calculated for the turning of the wheels. Figure 3.4 shows the concept and utilized a rack and pinion gear system for the steering of the wheels. The servo motor will rotate the gear to move the shaft and thus turning the platform left or right, and the differences in the gears ratio makes the angle of rotation bigger.

The usage of the caster wheels increases its maneuverability and makes the platform turn easier and smoothly. The problem with this design is the complexity of calculating and fabricating the right size of the gearing system to obtain the required turning angle of the system. The servo motor also needed to be powerful to withstand the load from the weight of the platform.

3.1.5 Design Concept #5

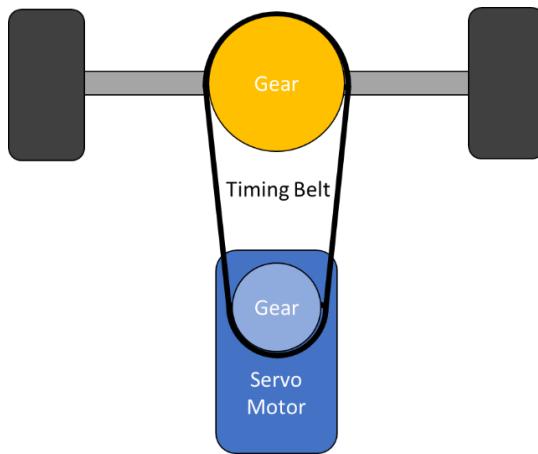


Figure 3.5: The sketching of Design Concept #5.

Based on the product in Chapter 2.5.4 Self-Driving Bicycle, the steering system is a simple belt/wire driven by a servo motor that helps it to turn the handles. This can be integrated into our VRU soft target platform as one of the methods of steering by connecting the gearing system into the servo motor and the shaft of the wheels.

The connection can be utilized either a timing belt or chain, with both having a different advantage and disadvantage. For instance, a belt driven system has lower maintenance and light-weight but it is easier for slipping to occur and the belt is vulnerable to wear and tear. Meanwhile, a chain driven system has a higher torque transfer and non-slip drive condition, but requires maintenance and lubrication, in addition to the noise of the metal connection system.

3.1.6 Design Concept #6

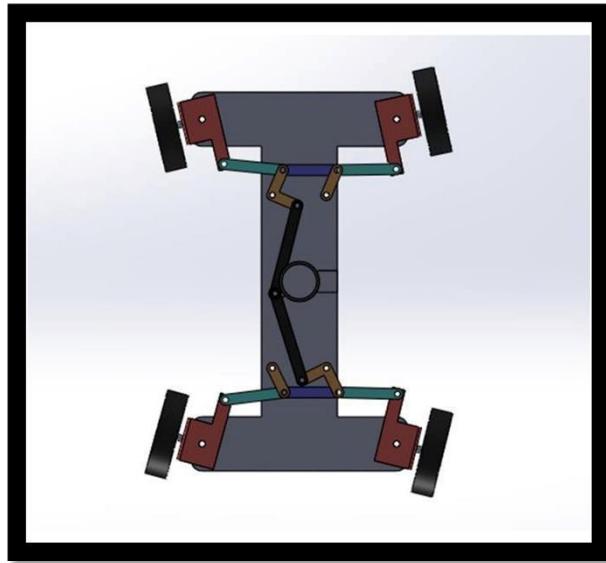


Figure 3.6: The sketching of Design Concept #6.

This concept utilized a single servo motor to drive a 4-wheel steering system. With high maneuverability and robustness, this system is a reliable concept to be used as the design of the platform. The design is also more powerful and able to withstand high loads and move in different types of road conditions. However, the complexity made it difficult to fabricate in a short time and the fabrication cost is higher than the given budget of the project.

3.1.7 Design Concept #7

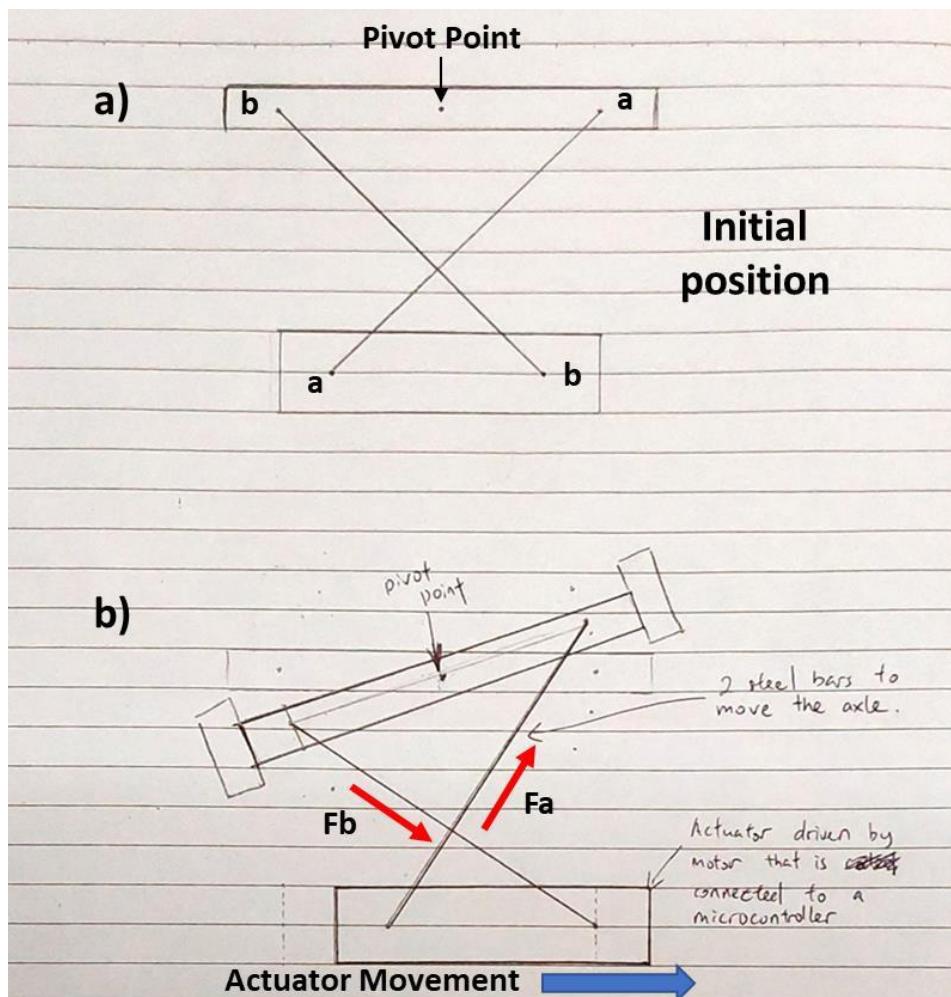


Figure 3.7: The sketching of Design Concept #7.

This design concept uses an actuator driven by motor that is connected by a microcontroller. Based on Figure 3.7, we can observe both beams, namely beam a-a and beam b-b, connected to the wheel axle and the actuator in a cross shape during initial position. When the actuator moves to the right direction, we can see the beam a-a has force that pushes the axle, while the force in beam b-b pulls the axle, making it turn to the left side.

Unfortunately, due to the absence of related existing products as reference, the system have a lower reliability to be successful and making it more difficult and time consuming to be fabricated.

3.1.8 Controller Design

A controller needs to be designed for the projects that enable the user to control the forward movement, the speed, the turning of left and right, and the preset route of the platform easily. With the current system has already an embedded coding on the microcontroller (ESC) system, we have developed a new controller that will be integrated with the current controller system as in Figure 3.8.

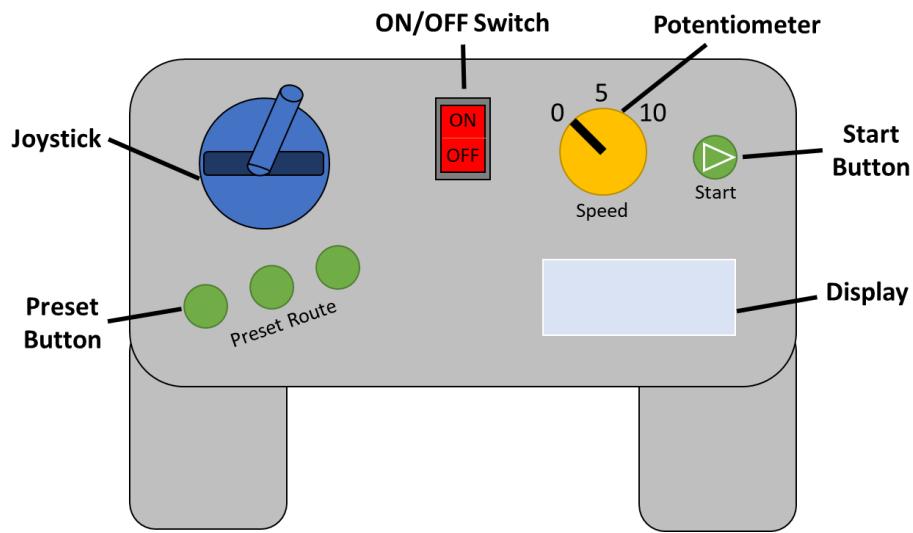


Figure 3.8: The sketching of controller design concept.

The design will utilize two microcontrollers, namely the ESC for the movement and Arduino for the steering system. The idea of the system is that the speed can be controlled by using a potentiometer connected to the ESC board. The user will be able to choose the speed of 5km/h or 10km/h depending on the situation needs. There are two inputs of the current flow to the potentiometer which are from the ESC board through a push start button and the platform will start moving as the speed is set and the button is pushed, and the other from the Arduino board through a relay for the preset route that will be explained later. The steering system is controlled by using the joystick connected to the Arduino Uno. The signal from the system will be connected by using RF transceiver to the platform. The preset route also will be coded into the Arduino board to ease the programmed in the steering system. To make the embedded preset route coding less complex, we decided to only program on the steering system and send signal to the relay attached to the potentiometer to allow current to flow and move the platform.

Lastly, the board will be powered by a lithium-ion battery and can be either rechargeable or non-rechargeable, depending on the necessity of the projects. The sketching concept of the electrical circuit can be observed in Figure 3.9 below.

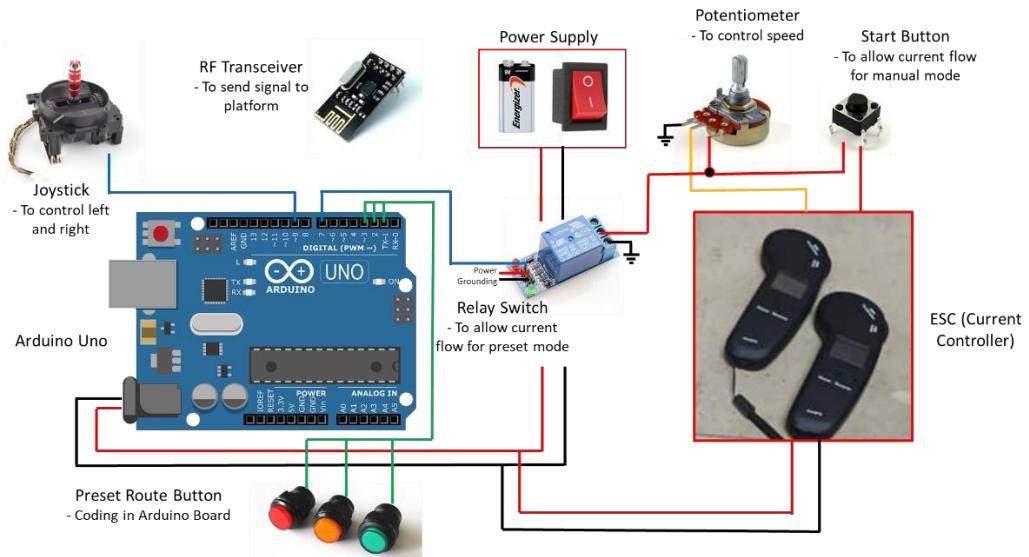


Figure 3.9: The concept of the electrical components in the controller design.

Another method for the controller is to design and fabricate from scratch without using the current ESC system. By starting everything from nothing, we were able to design by using only one microcontroller to ease the programming and control. Hence, we decided to use Arduino Mega as the brain of the system due to its bigger flash memory (256kB) compared to Arduino Uno (32kB), which allows a larger code to be programmed. The basis of the system will still be using the same concept as the integrated controller design.



Figure 3.10: The image of Arduino Mega 2560.

(Source: <https://store.arduino.cc/products/arduino-mega-2560-rev3>)

3.2 Design Evaluation and Selection

3.2.1 Evaluation Table

a) Movement

The movement criteria are used to evaluate which components are the best to ensure the platform moves smoothly. Based on table 3.1, it was clear that the DC motor with ESC is the best option to control the platform as it is already integrated with the current system and reliable for the projects. The main concern is on the preset speed condition, which can be overcome by using the potentiometer to control the speed.

	Weight Factor	DC Motor with ESC	Servo Motor	Stepper Motor
Ability to move forward	3	5 (15)	4 (12)	1 (3)
Ability to move backward	1	3 (3)	4 (4)	1 (1)
High torque	3	4 (12)	4 (12)	5 (15)
Preset route	3	5 (15)	3 (9)	3 (9)
Preset speed (5-10km/h)	3	3 (9)	1 (3)	1 (3)
Braking system	2	4 (8)	1 (2)	1 (2)
Total		62/75 x100% = 82.67%	42/75 x100% = 56.00%	33/75 x100% = 44.00%

Table 3.2.1: The evaluation table for the movement criteria.

b) Steering System

The steering system criteria focused heavily on the design concept of the platform that has been discussed back in Chapter 3.1. Upon evaluating all 7 design concepts available, we observed that Design 1 has the same score as Design 4. The deciding factor for these criteria is the complexity due to all the conditions with the weight factor of 3 having the same score. Design 1 is more complex due to the complex calculation required for the differential steering system and thus, Design 4 has been selected as the best steering system design.

	Weight Factor	Design 1	Design 2	Design 3	Design 4	Design 5	Design 6	Design 7
Maneuverability	3	5 (15)	5 (15)	4 (12)	5 (15)	5 (15)	5 (15)	3 (9)
Complexity	2	2 (4)	3 (6)	3 (6)	3 (6)	3 (6)	1 (2)	4 (8)
Traction Force	3	5 (15)	4 (12)	3 (9)	5 (15)	4 (12)	5 (15)	2 (6)
Robustness	3	4 (12)	4 (12)	3 (9)	4 (12)	3 (9)	5 (15)	3 (9)
Cost	2	5 (10)	4 (8)	4 (8)	4 (8)	3 (6)	1 (2)	4 (8)
Height(<85mm)	3	4 (12)	3 (9)	4 (12)	4 (12)	4 (12)	4 (12)	4 (12)
Weight	2	3 (6)	3 (6)	3 (6)	3 (6)	3 (6)	3 (6)	2 (4)
Maintenance	1	4 (4)	4 (4)	3 (3)	4 (4)	3 (3)	2 (2)	4 (4)
Total		78/95 x100% = 82.11%	72/95 x100% = 75.79%	65/95 x100% = 68.42%	78/95 x100% = 82.11%	69/95 x100% = 72.63%	69/95 x100% = 72.63%	60/95 x 100% = 63.16%

Table 3.2.2: The evaluation table for the steering system criteria.

c) Control System: Movement

Since the integrated controller used the current system controller, it is far less complex and proven to be reliable compared to fabricating a new control system from scratch using Arduino Mega, which required new program code and system to be developed.

	Weight Factor	Integrated (ESC)	From Scratch (Arduino)
Controllability	2	5 (10)	4 (8)
Connectivity	3	5 (15)	4 (12)
Range	3	4 (12)	4 (12)
Complexity: -Number of components -Program code -Assembly	2	4 (8) 5 (10) 4 (8)	4 (8) 2 (4) 3 (6)
Total		63/70 x100% = 90%	50/70 x100% = 71.43%

Table 3.2.3: The evaluation table for the control system (movement) criteria.

d) Control System: Steering

The score between both methods is the same due to the same process required in developing the steering control system using Arduino. However, the favor goes to the integrated design as the risk of error in coding will be less due to the usage of 2 microcontrollers, and the integrated design for the movement system is far better than starting from scratch.

	Weight Factor	Integrated (Arduino)	From Scratch (Arduino)
Controllability	3	4 (12)	4 (12)
Connectivity	3	5 (15)	5 (15)
Range	3	4 (12)	4 (12)
Complexity: -Number of components -Program code -Assembly	2	3 (6) 3 (6) 4 (8)	3 (6) 3 (6) 4 (8)
Total		59/75 x100% = 78.67%	59/75 x100% = 78.67%

Table 3.2.4: The evaluation table for the control system (steering) criteria.

3.2.2 Finalized Design

The platform in Figure 3.11 has been designed based on Design Concept #4, by changing the front wheels with caster wheels and using a rack and pinion gearing system connected to a servo motor as the steering system, while Figure 3.12 shows the 3D rendering for the design concept of the controller.

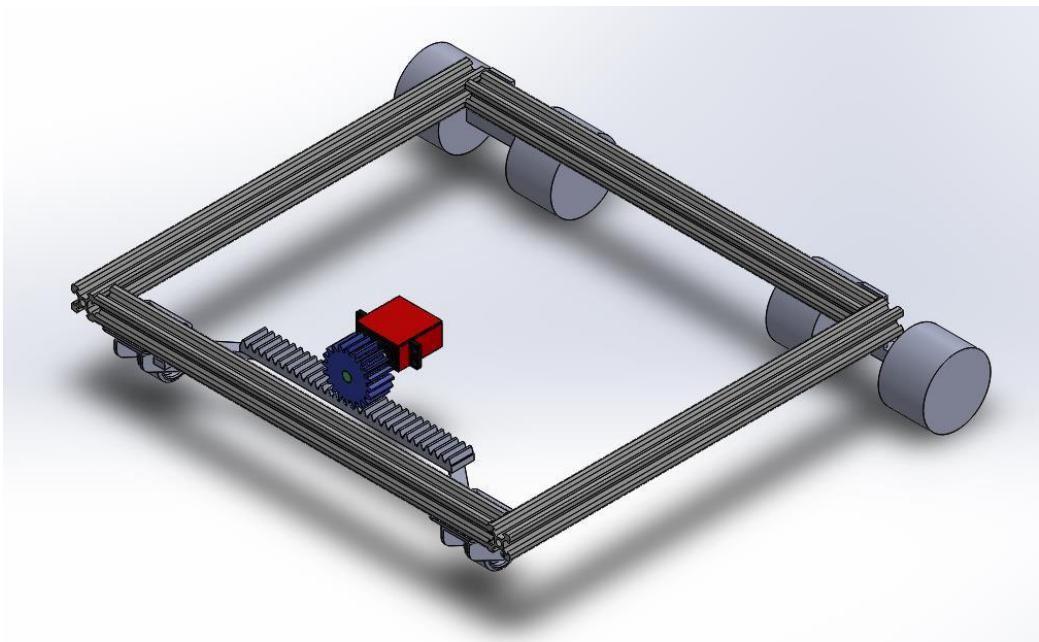


Figure 3.11: The 3D design of the platform.

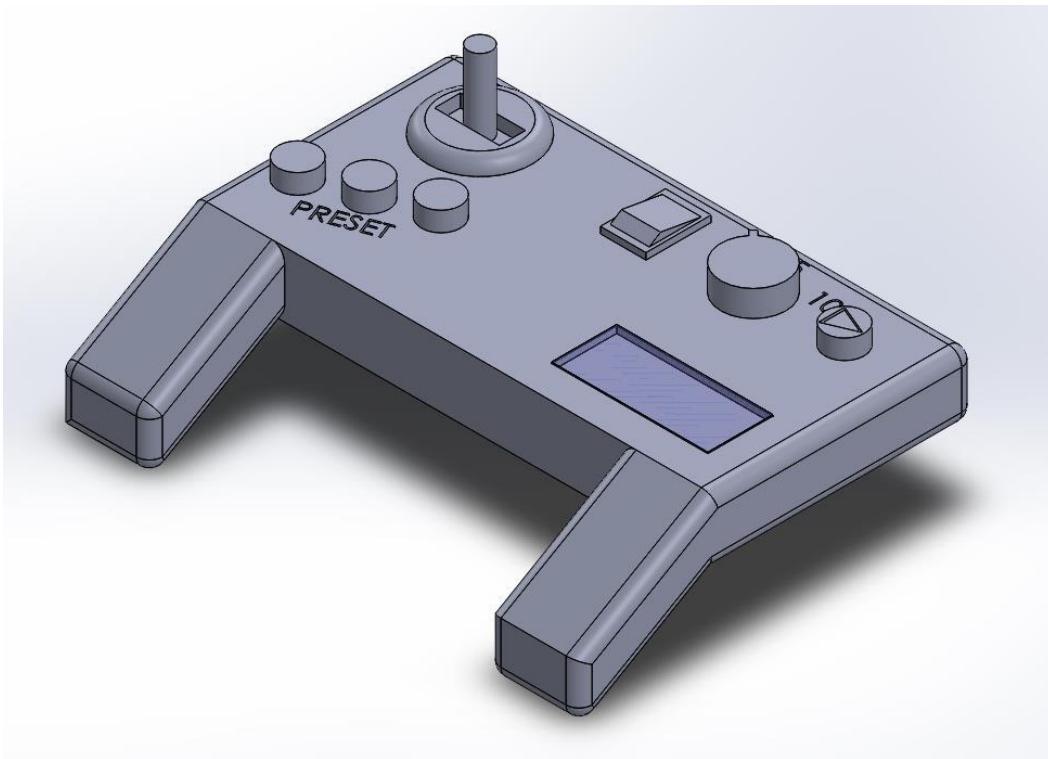


Figure 3.12: The 3D design of the controller.

3.3 Requirements

To develop a test platform for VRU soft target. A soft target will be placed above the platform which will then be tested on a track. The height of the developed platform should adhere to a

guideline from Euro NCAP. The platform should be able have the durability to withstand an impact from a car during testing, such as being run over.

3.3 Concept

3.3.1 Initial design

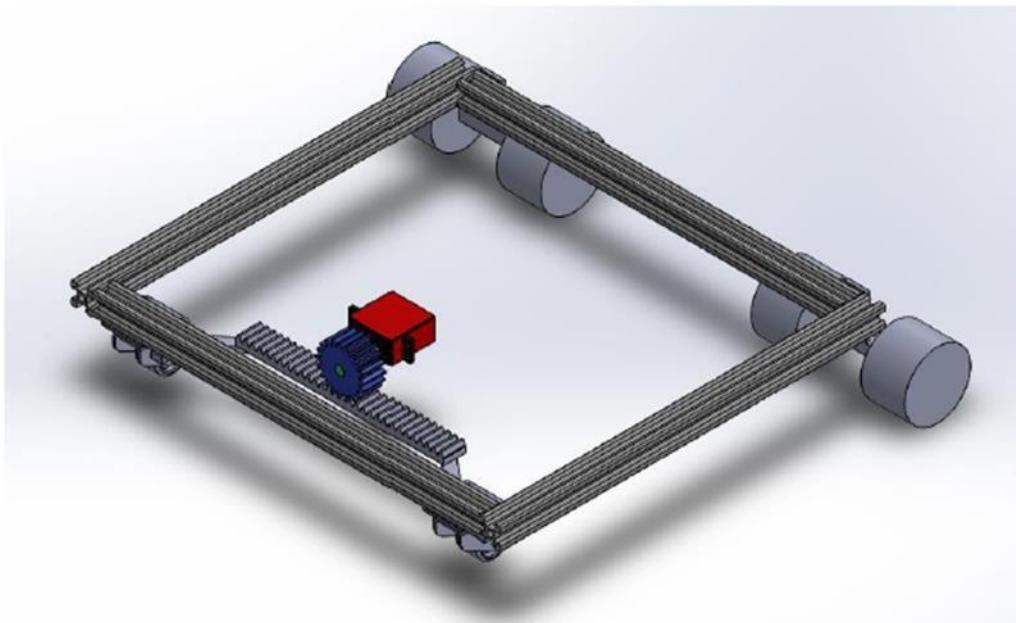


Figure 3.13: 3D rendering of our initial design

Our initial design used caster wheels as the front wheel and a rack and pinion gearing system that is connected to a servo motor as the steering system. Figure above shows the 3D rendering of the design.

3.3.2 New Design

* 3d model of new design.

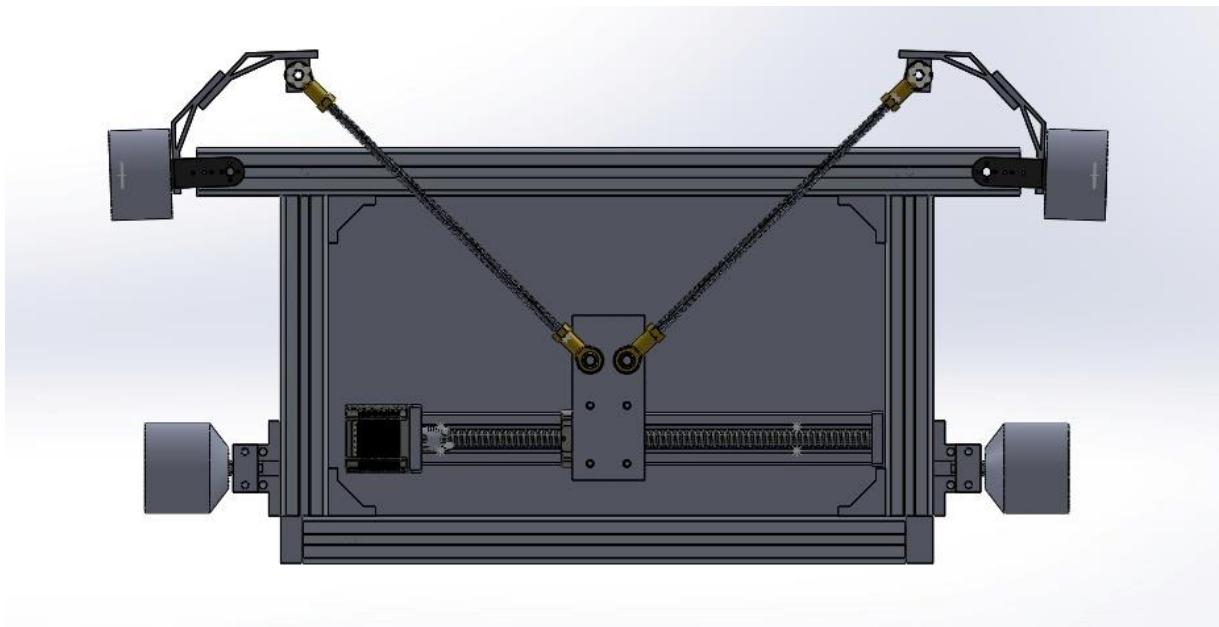


Figure 3.14: 3D model of our new design.

Our new design uses an actuator driven by a stepper motor, connected to tie rods which are connected to the knuckle which holds the wheels. The actuator moves left or right which turns the wheels which will be our steering system. The platform is driven by 2 motors from an electric skateboard motor.

3.3.3 Concept evaluation

In order to evaluate the different concepts, an evaluation matrix is required to weigh the different criteria.

0 = not significant 1 = even 2 = significant	Manufacturing	Cost	Assembly	Durability	Construction	Sum	Weight
Manufacturing	1	0	2	0	1	4	0.44
Cost	1	1	2	1	2	7	0.78
Assembly	0	0	1	0	2	3	0.33
Durability	2	1	2	1	2	8	0.89
Construction	1	0	0	0	1	2	0.22

Table 3.3.1: Evaluation matrix

The cost should be kept as low as possible to be within budget. The durability is the most important criterion since it should be durable enough to withstand an impact from a car. The construction of the platform is the least significant.

Evaluation

	Weight (1-5)	Concept 1	Concept 2	Concept 3	Concept 4	Concept 5	Concept 6
Manufacturing	0.44	4	3	3	3	3	2
Cost	0.78	4	2	2	3	2	1
Assembly	0.33	4	2	2	3	3	2
Durability	0.89	1	3	2	5	4	5
Construction	0.22	3	2	2	2	2	1
Total Score		7.75	6.65	5.7	9.54	7.87	6.99
Value		0.58	0.50	0.43	0.72	0.59	0.53

Table 3.3.2: Evaluation table

The value in the table refers to the variant evaluation according to VDI on technical and economic value where 0.1 = not satisfying to 1.0 = ideal

Value scale	
0.1	Unsatisfying
0.2	
0.3	
0.4	Sufficient
0.5	
0.6	Good
0.7	
0.8	Very good
0.9	
1.0	Ideal

Table 3.3.3: Value Scale

3.3.4 Evaluation

	Weight (1-5)	Concept 1	Concept 2	Concept 3	Concept 4	Concept 5	Concept 6
Manufacturing	0.44	4	3	3	3	3	2
Cost	0.78	4	2	2	3	2	1
Assembly	0.33	4	2	2	3	3	2
Durability	0.89	1	3	2	5	4	5
Construction	0.22	3	2	2	2	2	1
Total Score		7.75	6.65	5.7	9.54	7.87	6.99
Value		0.58	0.50	0.43	0.72	0.59	0.53

Table 3.3.4: Evaluation table

3.3.5 Reasons for design change

Some improvements have been made from the initial design. In our initial design, we used a rack and pinion gearing system which we found to be difficult to assemble onto the platform due resource shortage and difficulty in meeting the guidelines. Our new design is more robust compared to the initial design due to its stronger materials used such as the metal tie rod and the actuator to turn the wheels.

3.3.6 Problems encountered and solutions

During our assembly and testing we have encountered some problems. The stepper motor which drives the actuator was not strong enough to move the heavy platform so we had to use a bigger motor which a used and old motor. It was damaged and we took some time to find replacements for the parts.

The actuator also had some damage in its bearings which caused the actuator to not move and we had to replace it. Once we finished assembling the actuator, we had to measure the tie rods which are too long and we had to cut it to the exact length we need to keep the wheels straight.

We also noticed the 2 driving motors that is attached to the platform frame by a bracket was crooked due to the frame not being straight when we assembled it which caused the wheels to

have camber and not have enough traction when driven. We had to disassemble the frame and reassemble it to ensure it is parallel to the ground.

3.5 Assembly

For assembly, we use 4040 Aluminium profile as our chassis platform with a galvanized iron plate for our base to protect the inner electrical circuits and boards from small debris on the road.

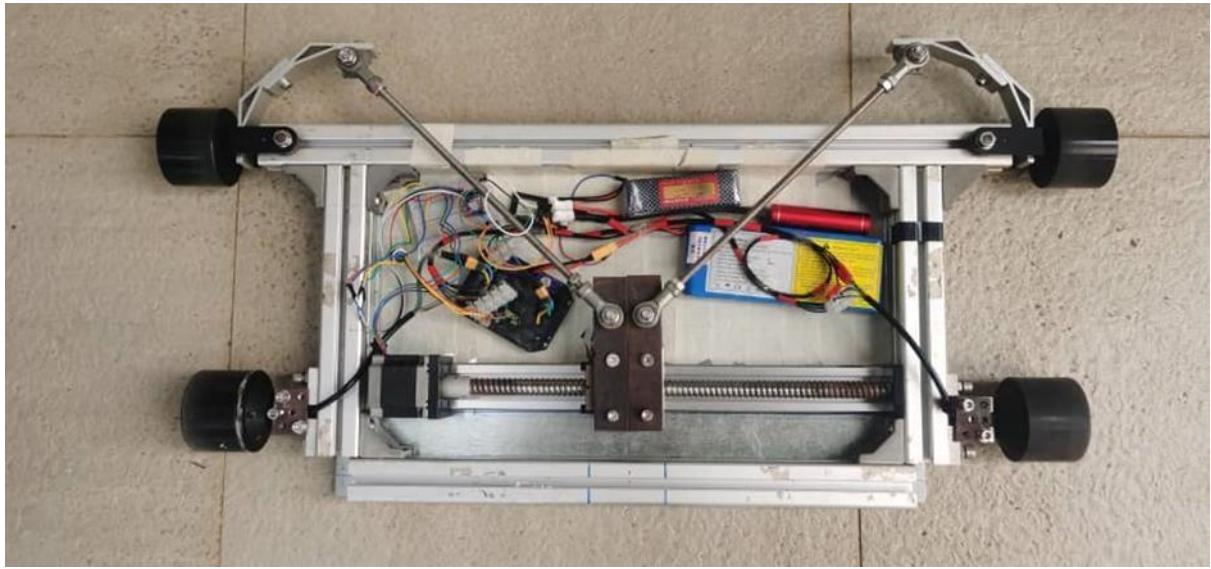


Figure 3.15: Our new design.

For securing drive motor, we first assemble the aluminium block bracket to the chassis with 2 bolts and nuts. And then we secure the drive motor to the bracket by using cover plate on top of the bracket with screws on each side.

As for steering systems, we created bracket to connect the wheel in order to maneuver our platform without compromising the height of our model by using C bracket and tie rod to connect to the steering system controlled by stepper motor. For both top and bottom of the connection at the bracket, we included bearing to ensure the smoothness of the maneuverability, however part of the aluminium profile needed to be cut out to let bearing sitting inside the aluminium profile to satisfy the height of the platform which below 85 mm by Euro NCAP.

3.6 Electrical Parts

3.6.1 Drive Motor

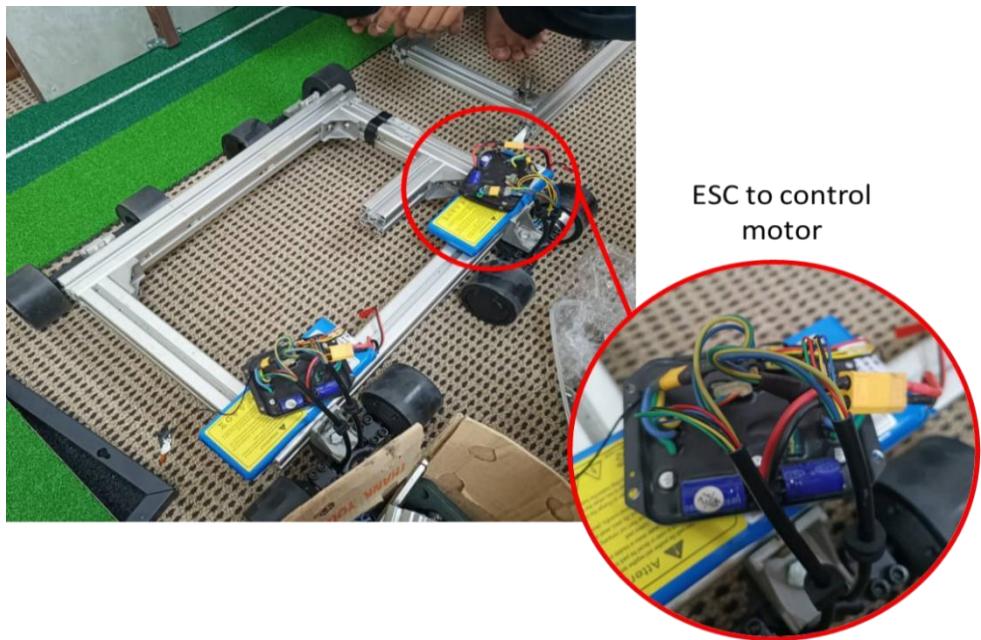


Figure 3.16: Electrical system on existing product.

Figure 3.16 shows the electrical system that uses two Electronic Speed Controller (ESC) to control the motors which is a brushless DC motor, and drive the platform. The motors came with built in ESC and its own electrical system. The project utilized a brushless DC motor with a built-in ESC and electrical system, offering several advantages, including enhanced efficiency and reliability. The integrated system eliminated the need for any modifications, as it was explicitly designed to function optimally without alterations. Additionally, this approach minimized potential issues that could arise from tampering with the motor's internal setup, ensuring a safer and more straightforward implementation.

Considering the significant cost associated with the brushless DC motor, the decision to utilize it as the primary propulsion system was embraced by the project. The investment in this high-quality motor underscored its superior performance and long-term durability, making it a prudent choice for our platform design.

Given the motor's self-contained nature, the focus of the project was on optimizing its integration into the platform's chassis. The approach involved ensuring the secure mounting of the motor onto the platform and establishing reliable connections between the motor and the

it's electrical components. The motor's ESC capabilities allowed seamless communication with the RC transmitter, enabling precise speed control and smooth maneuverability.

Throughout the project, manufacturer guidelines were respected, and the motor and its built-in ESC were adhered to strictly as supplied. This decision proved to be a strategic choice, as it not only enhanced the efficiency and performance of the platform but also mitigated potential risks associated with unauthorized modifications.



Figure 3.17: Electric Skateboard from Anzo company.

(<https://anzoskate.com/products/a1-us-warehouse-stock-glass-fiber-217000-lithium-battery-10s2p-600w-2-wheel-hub-motor-new-electric-skateboard>)

The wheels set utilized in the project, as shown in Figure 3.17, originates from Shandong Anzo International Trade Company Limited, a prominent manufacturer and trading company established in 2019, located in Shandong, China. Specializing in the production of high-quality electric skateboards, Anzo has gained recognition for its cutting-edge designs and advanced technologies in the electric mobility industry.

The electric skateboard wheels provided by Anzo exemplify the company's commitment to excellence, incorporating the latest advancements in materials and engineering. These wheels are specifically designed to deliver superior performance, durability, and enhanced riding experiences. Their expertise in electric skateboard manufacturing ensures that the wheels meet the highest standards of quality and safety. By sourcing the wheels from Anzo,

the project benefits from the company's industry expertise and dedication to innovation. The use of Anzo's electric skateboard wheels guarantees optimal performance and reliability, adding value to the overall design of our project.

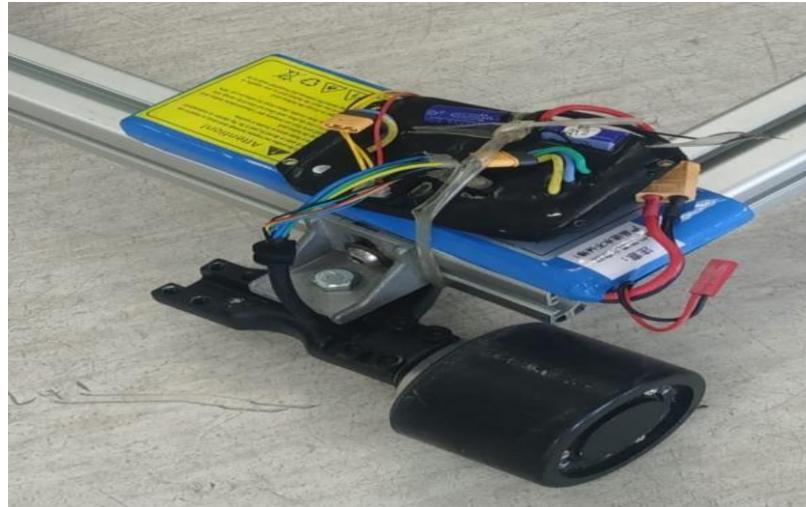


Figure 3.18: Wheels set from skateboard.

The wheels set utilized in the project, as shown in Figure 3.18, is sourced from Shandong Anzo International Trade Company Limited. Although specific details about the wheel's electrical system were not provided by the company, it was established that the wheels are powered by a 600W BLDC (Brushless Direct Current) motor. Control of the wheels is achieved through a wireless controller, offering three distinct speed modes: low (5 km/h), medium (25 km/h), and high (50 km/h). The availability of different speed settings allows for versatile usage, catering to various user preferences and different terrains.

The primary source of power for the electric skateboard's wheels is a 10S2P-36V battery, featuring a dynamic lithium-ion composition. The battery boasts a capacity of 9.6AH, which corresponds to 9600mAh. With this battery configuration, the electric skateboard can cover a remarkable distance of up to 30-40 km on a single charge. This impressive range ensures extended ride times, making it ideal for both short trips and more extended journeys.

Incorporating the 10S2P-36V lithium battery further enhances the project's performance and efficiency, providing a reliable and long-lasting power source for the electric skateboard. The battery's dynamic nature ensures a consistent supply of energy to the 600W BLDC motor, enabling smooth and seamless acceleration across the available speed modes.



Figure 3.19: ESC used in our project.

Figure 3.19 shows the Electronic Speed Controller (ESC), which plays a critical role in the operation and performance of our project. As an integral component of the powertrain system, the ESC serves as the interface between the motor, battery, and the controller's inputs, facilitating smooth and precise speed regulation. The significance and functionality of the ESC in our project will be elaborated as below:

- a. Motor Control: The primary function of the ESC is to control our project's motor speed and direction. By receiving input signals from the wireless controller, the ESC adjusts the motor's power output accordingly. When the throttle or acceleration is increased, the ESC increases the power delivered to the motor, propelling our platform forward. Similarly, reducing the throttle or activating the brake on the wireless controller prompts the ESC to control the motor's speed and initiate deceleration or reverse motion.
- b. Smooth Acceleration and Braking: The ESC is equipped with sophisticated algorithms that ensure smooth acceleration and braking. Rapid and jerky changes in speed can be unsafe. The ESC's precise control over the motor's power output enables gradual acceleration and deceleration, providing a seamless and secure control for our platform.
- c. Regenerative Braking: Our project features regenerative braking, and the ESC is responsible for managing this process. When the brake is applied, the ESC reverses the motor's rotation, turning it into a generator. This regenerative braking system converts the kinetic energy of the motion back into electrical energy, which is then fed back into the battery, partially recharging it and extending the platform's range.
- d. Safety Features: The ESC incorporates several safety features to protect the components of the platform. These safety measures may include over-current protection,

overtemperature protection, and low-voltage protection. If any of these conditions are detected, the ESC may limit the power output or initiate a safety shutdown to prevent damage to the motor, battery, or other electrical components.

- e. Real-Time Data Monitoring: ESCs provide real-time data monitoring and feedback, allowing us to keep track of various performance metrics such as battery voltage, motor temperature, and current draw. This information empowers us to make informed decisions and helps optimize battery usage for the platform.



Figure 3.20: BLDC motor used by platform.

The motor utilized in our project, as shown in Figure 3.20, is a brushless DC motor with a power output of 600W, capable of achieving a maximum speed of 45-50 km/h. The wheels used have dimensions of 85*59mm and are made of PU material. The motor's maximum load capacity is 150kg.

The brushless DC motor plays a crucial role in our project, providing the necessary propulsion for the platform. With its high-power output of 600W, the motor delivers impressive performance, enabling the platform to reach a maximum speed of 45-50 km/h. The 85*59mm wheel size, along with the PU material, enhances the overall movement quality and grip of the platform. PU wheels are known for their durability and resistance to wear, ensuring longevity and consistent performance during extended use. Additionally, the PU material contributes to a smooth and comfortable ride, absorbing shocks and vibrations from uneven surfaces, further enhancing the motion of the platform.

The motor's maximum load capacity of 150kg ensures that our project can accommodate a dummy which is much lighter. This feature enhances the platform's versatility and usability, making it suitable for various VRU conditions on the road.

The steering system was designed and constructed by utilizing a stepper motor with a linear actuator, creating a connection to enable the rotation of both front wheels through two rods. For facilitating communication between the controller and the stepper motor, two ESP32 boards were employed in the setup. Additionally, the controller, which was equipped with a joystick, was responsible for controlling the direction of the stepper motor.

In this passive configuration, precise and responsive steering control was achieved, contributing to a seamless driving experience. The stepper motor was connected to a 12V battery, which served as its power source to enable its operation effectively.

Overall, the integration of the stepper motor, linear actuator, rods, ESP32 boards, and the joystick-equipped controller demonstrated a well-orchestrated system that provided reliable and efficient steering functionality for the vehicle.

3.6.2 Steering

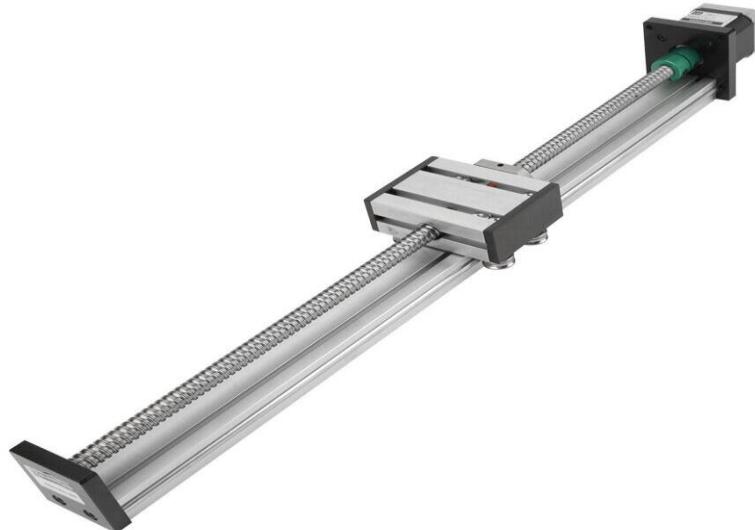


Figure 3.21: Stepper motor with linear actuator.

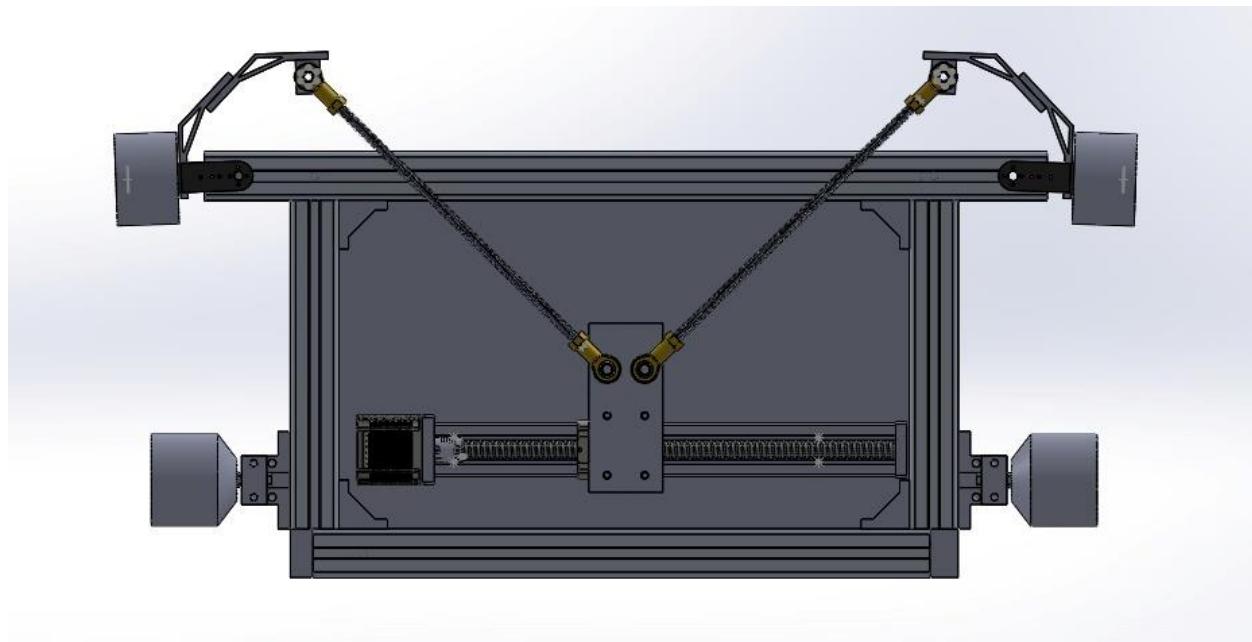


Figure 3.22: Design Model of the steering wheel

Figure 3.22 shows the stepper motor which is connected to the linear actuator. This mechanism is used as the steering system of our platform.

The linear actuator is made from durable metal, ensuring a sturdy and long-lasting construction that can withstand various operating conditions. The design and manufacturing process of the stepper motor and linear actuator prioritize high precision and meticulous craftsmanship. This is important for achieving accurate and consistent linear motion. The combination of materials and design contributes to a long operational lifespan for the linear actuator. The low coefficient of friction indicates that the components have been engineered to minimize friction, resulting in smoother and more efficient linear motion. [29]

The versatility of the stepper motor and linear actuator suggests that they can be used in a wide range of applications across different industries. The primary function of this system is to provide linear motion, meaning the output is a controlled and precise movement along a straight path. The linear shaft diameter is 12mm (0.47 inches). This dimension indicates the size of the shaft along which the linear motion occurs. The effective stroke is either 100mm (3.9 inches) or 400mm (15.75 inches). This refers to the maximum distance the actuator can move along its linear path. Stepper motors with linear actuators are commonly used in automation, robotics, CNC machinery, 3D printers, medical equipment, and other precision-controlled systems that require accurate linear positioning. The specifications of the linear actuator are as Table 3.6.2.1 below. [29]

Table 3.6.2.1 : Specifications of the linear actuator used. [30]

Material	Aluminium
Boundary dimension	40mm
Shaft diameter	12mm / 0.47inch
Rated horizontal load	60kg

Sliding block	60 x 80 x 30mm / 2.36 x 3.15 x 1.18inch
Screw	M6
Rated vertical load	18kg
Accuracy	0.03–0.05mm
Effective stroke	400mm / 15.75inch
Ball screw length	465mmLength: 57cm / 22.4in
Pitch-row length	60mm
Slide table size	490 x 40 x 20mm / 19.29 x 1.57 x 0.79inch
Weight	Approx. 2075g

3.6.2.3 Steering Assembly

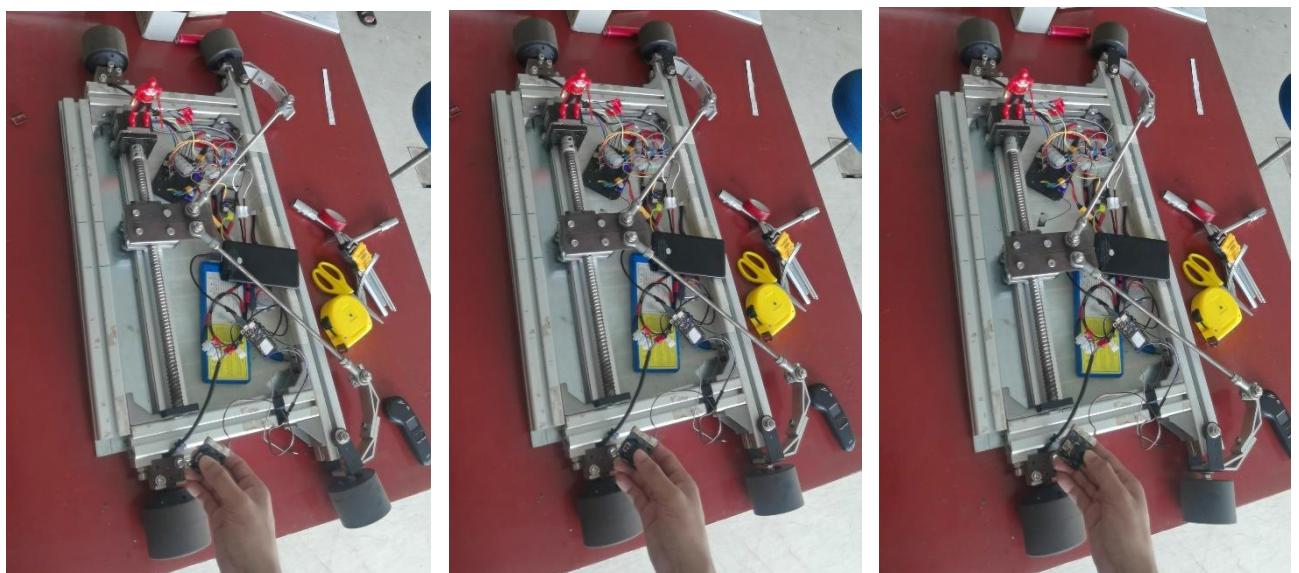


Figure 3.23: The complete assembly of the steering system when a. turning left, b. at the middle, c. turning right.

Figure 3.23 shows the complete assembly of the steering system of the platform. The steering system design revolves around the precise control of the front wheel steering angle using a combination of a stepper motor and a linear actuator. The intention is to achieve accurate and responsive steering adjustments for the platform. The central component of the system is the stepper motor. This motor serves as the driving force for the steering mechanism, converting electrical pulses into controlled rotational motion. Connected to the stepper motor, the linear actuator plays a crucial role in translating the rotational motion of the motor into linear movement. This linear motion is transferred to a sliding block along a predetermined path. The sliding block is guided by the linear actuator's path. As the linear actuator extends and retracts, the sliding block follows the same linear trajectory. The sliding block is affixed to rod end bearings, providing a pivoting point for the mechanical linkage to the front wheel brackets. These bearings facilitate

smooth rotation and movement of the steering system. Connected to the rod end bearings, the front wheel brackets hold the front wheels in place. The motion transmitted from the sliding block to the rod end bearings induces steering movement by altering the angle of the front wheels.

In summary, this innovative steering system design harnesses the power of a stepper motor and a linear actuator to achieve precise and controlled steering adjustments. The linear-to-rotational motion conversion process enables responsive changes in the front wheel angles, contributing to the vehicle's overall maneuverability and handling characteristics. This design can greatly enhance the precision and versatility of steering control of the platform.

3.6.3 Improvement

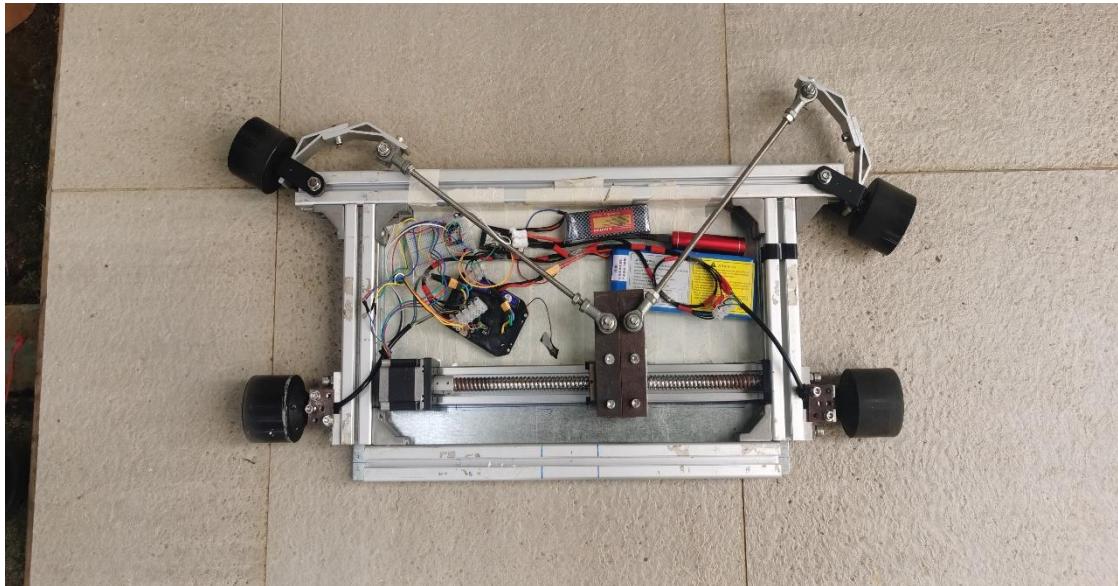


Figure 3.24: Same steer angle at both wheels during cornering.

While innovative in this steering mechanism approach, it on the other hand introduces a critical challenge that pertains to the foundational principles of steering mechanics, specifically the Ackermann steering geometry. The potential issue stems from the fact that both front wheels rotate at the same steering angle as shown in Figure 3.24, which directly contradicts the fundamental tenets of Ackermann steering law. This deviation from the Ackermann principle can have significant consequences, particularly in the form of wheel slip and compromised handling dynamics.

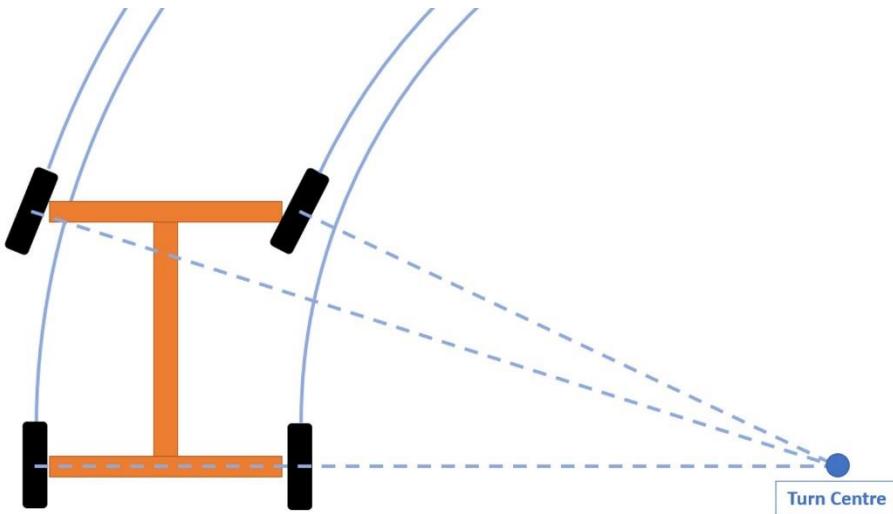


Figure 3.25: Ackermann steering geometry.
[\(https://racecar-engineering.telegraph.co.uk/articles/tech-explained-ackermann-steering-geometry/ \)](https://racecar-engineering.telegraph.co.uk/articles/tech-explained-ackermann-steering-geometry/)

Figure 3.25 shows the Ackermann steering system, which is a fundamental principle in vehicle steering geometry that ensures proper wheel alignment and turning angles for vehicles with multiple wheels, such as cars. Named after the 18th-century German engineer Rudolph Ackermann, this principle is designed to optimize the turning behavior of a vehicle, especially during cornering, by minimizing tire scrubbing and ensuring stable and efficient turning. [31]

The core idea behind Ackermann steering is that when a vehicle turns, the inside wheel (the wheel on the inner side of the turn) should follow a tighter turning radius compared to the outside wheel (the wheel on the outer side of the turn). This differential in turning angles between the two front wheels helps the vehicle maintain its intended path and reduces the stress on the tires, resulting in smoother and more controlled cornering. [31]

In an Ackermann steering system: [32]

Inner Wheel Angle: The inner front wheel is turned at a sharper angle than the outer front wheel. This allows the inner wheel to follow a tighter turning radius, accommodating the shorter path it needs to traverse during a turn.

Outer Wheel Angle: The outer front wheel is turned at a wider angle than the inner wheel. This enables the outer wheel to cover a larger distance during the turn, matching the trajectory of the vehicle.

Steering Linkage and Geometry: The steering linkage is designed to achieve these differential turning angles. The pivot points and link lengths are carefully calculated to ensure that the Ackermann principle is followed.

The benefits of Ackermann steering include reduced tire wear, minimized tire scrubbing (lateral movement of the tires), improved stability during cornering, and enhanced overall vehicle handling. Proper Ackermann geometry helps maintain consistent tire contact with the road surface, which is crucial for optimal traction and control. [32]

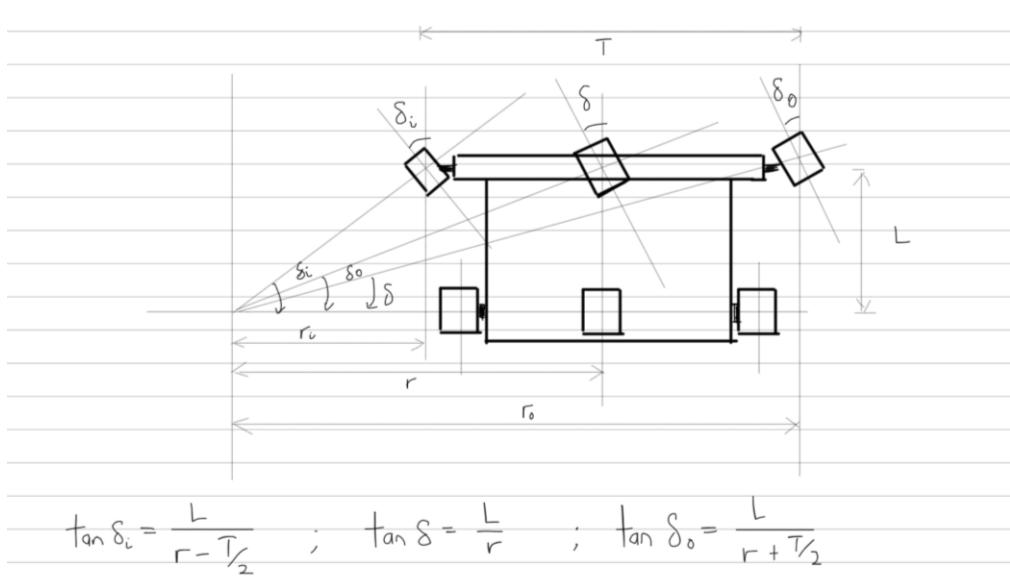


Figure 3.26: Ackermann steering system if applied to the platform.

As shown in Figure 3.26, the performance of the steering system of the platform can be greatly improved by implementing the Ackermann's steering system. Implementing an Ackermann steering system into the project offers substantial benefits in terms of enhanced cornering performance, reduced tire wear, and improved stability. However, this enhancement does come with the trade-off of increased complexity in the design and control of the steering mechanism. [33]

Advantages of Ackermann Steering:

Optimal Cornering Dynamics: By following the Ackermann steering principle, the platform's cornering dynamics will be significantly improved. Each wheel will be precisely aligned to its intended path, reducing the likelihood of tire scrubbing and allowing for smoother and more efficient turns.

Minimized Tire Wear: Uneven tire wear, which can occur due to tire scrubbing in non-Ackermann steering systems, will be greatly mitigated. This means that the tires will last longer, resulting in cost savings and consistent performance.

Enhanced Stability: Ackermann steering contributes to overall vehicle stability during cornering. The optimized turning angles help maintain the platform's intended trajectory and prevent instability or oversteering. This is very important as the platform will carry a dummy in an ADAS test.

Challenges and Complexity:

Differential Steer Angles: As noted before, the inner wheel's steering angle must be sharper than that of the outer wheel in an Ackermann system. This necessitates careful design of the steering linkage, including pivot points and link lengths, to achieve the desired angles.

Mechanical Linkage Design: Designing a steering linkage that accommodates the differential steering angles can indeed be more complex. This requires accurate calculations and precision in manufacturing to ensure that the angles are achieved accurately.

Control System Complexity: Implementing an Ackermann steering system often involves more sophisticated control algorithms and sensors to ensure that the correct angles are maintained during turns. This complexity extends to the software that interfaces with the stepper motor and linear actuator.

Manufacturing and Assembly: The precise construction and assembly of the mechanical components become crucial in achieving accurate steering angles. Tolerance and alignment need to be meticulously managed.

In summary, while an Ackermann steering system may introduce increased complexity, the benefits in terms of cornering performance and tire wear reduction can justify the effort. By embracing the principles of Ackermann steering, it can offer enhanced handling and control, elevating its performance to a higher level. [33]

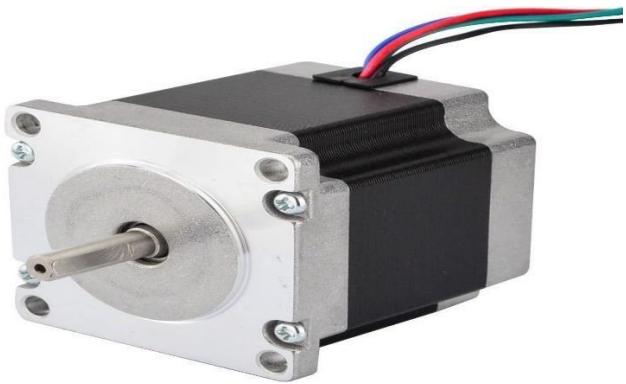


Figure 3.27: NEMA 23 stepper motor.

NEMA 23 stepper motor, as shown in Figure 3.27, is a standard sizing classification for stepper motors defined by the National Electrical Manufacturers Association (NEMA). The "23" in NEMA 23 represents the faceplate size of the motor, which is approximately 2.3 inches by 2.3 inches (57 mm x 57 mm). NEMA 23 stepper motors are commonly used in various applications, including CNC machines, 3D printers, robotics, and automation systems, due to their moderate size and torque capabilities. Operation: Stepper motors operate by converting electrical pulses into discrete mechanical movements, where each pulse corresponds to a step, resulting in precise position control. They move in fixed angular increments (e.g., 1.8 degrees per step for a typical NEMA 23 motor). Stepper motors are usually operated in an open-loop control system, meaning they do not require feedback for position control, which simplifies the control system. They are relatively straightforward to control and don't necessitate complex control algorithms, making them easy to use in various applications. Stepper motors are generally more cost-effective than servo motors, which can be beneficial for budget constrained projects. Stepper motors provide substantial holding torque even when stationary, allowing them to hold position without the need for additional holding brakes. Stepper motors can generate more heat during operation, especially at higher speeds, which may impact their continuous duty performance. Here are some pros and cons of using stepper motor in our project:

Pros:

- Precise Positioning: Stepper motors offer precise positioning and repeatability due to their fixed-step movement.
- No Feedback Required: Stepper motors don't require expensive feedback devices like encoders, reducing overall costs.

- Easy to Use: They have a simple control scheme and are suitable for applications that don't require high-speed or high-torque performance.

Cons:

- Limited High-Speed Performance: Stepper motors might struggle at higher speeds due to their discrete step movement, leading to a phenomenon called "mid-range instability."
- No Velocity Control: Since they lack feedback, stepper motors don't inherently offer velocity control or smooth motion profiles.
- Missed Steps: In certain conditions, stepper motors can lose synchronization and miss steps, causing position errors.

Below are the details of the NEMA23 stepper motor used in our project:

Table 3.6.3: NEMA23 specifications.

Model	57BYGH56
No. of Phase	2 Phase 4 wires
Rated Phase Current	2.8A
Phase Resistance	0.9Ω
Phase Inductance	4.7mH
Step Angle	1.8 Degree
Shaft	6.35 D-Shape
Rated Voltage	12V – 24V DC
Holding Torque	1.26N.m.
Shaft Radial Force	10kg/cm
Wire length	100cm
Frame size	56mm x 56mm
Motor Body Length L _{MAX}	56mm
Weight	770grams

Connection Diagram:

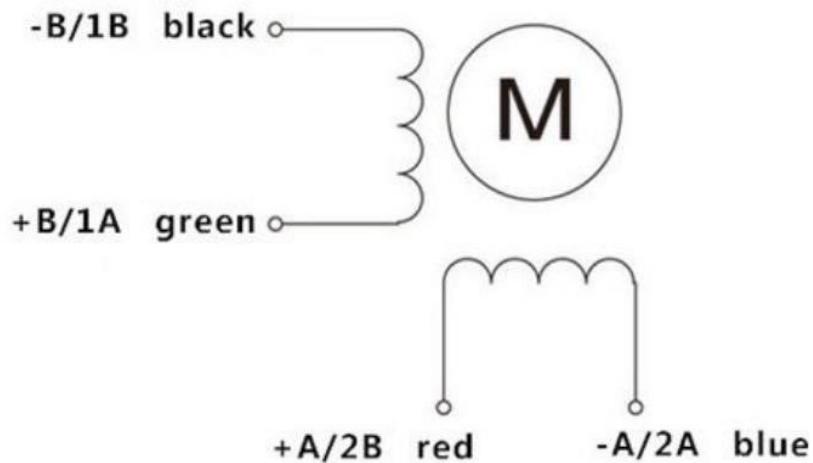


Figure 3.28: Connection diagram of a stepper motor.

The four-color wires on a stepper motor typically indicate the individual coil windings and their connections. The specific colors and their corresponding connections can vary between different stepper motor models, so it's crucial to refer to the motor's datasheet or documentation for accurate information.

However, in many cases, the four-color wires are as follows:

- Black Wire: Usually, the black wire corresponds to one end of the first coil winding (Coil A).
- Green Wire: The green wire typically represents the other end of the first coil winding (Coil A).
- Red Wire: The red wire often corresponds to one end of the second coil winding (Coil B).
- Blue Wire: Finally, the blue wire is commonly associated with the other end of the second coil winding (Coil B).

Unit: mm

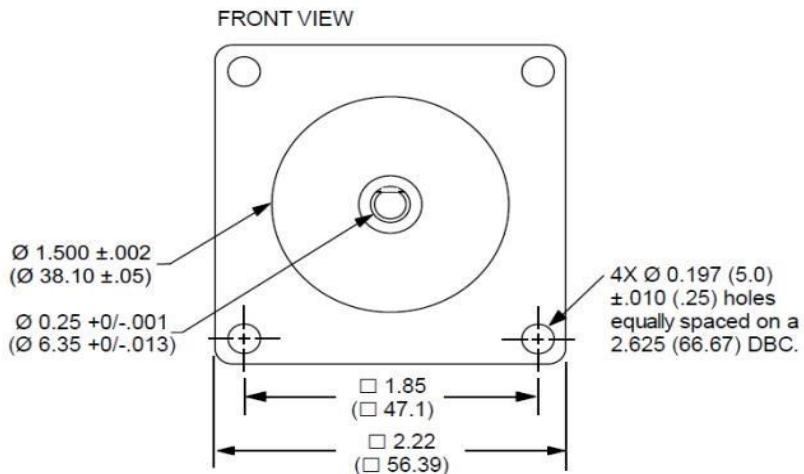
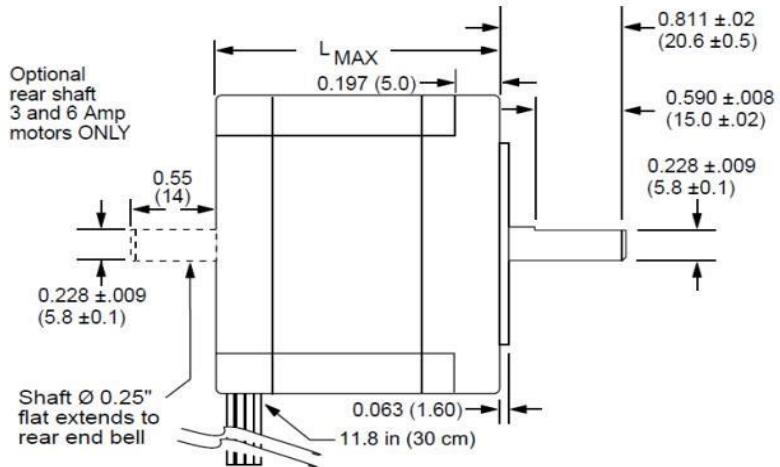


Figure 3.29: Mechanical dimension of the stepper motor.

Figure 3.29 shows the mechanical dimension of the stepper motor. It is worth noting that the physical size of a stepper motor is directly related to its torque output. Larger stepper motors generally have higher torque capabilities compared to smaller ones. If an application requires high torque like our platform to move a load or overcome resistance, a larger stepper motor may be more appropriate. Then, larger stepper motors have more surface area for heat dissipation, which can be beneficial for the platform requiring extended operation at high currents. Smaller motors might heat up more quickly and might need additional cooling mechanisms.

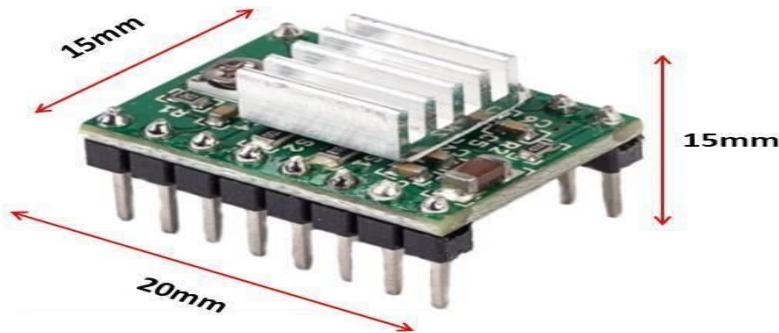


Figure 3.30: A4988 stepper motor driver.

Figure 3.30 shows the A4988 stepper motor driver used to control the stepper motor. The A4988 is a popular and widely used stepper motor driver integrated circuit (IC) designed for driving bipolar stepper motors. It is commonly used in various DIY projects, 3D printers, CNC machines, robotic systems, and other applications that require precise control of stepper motors. The A4988 stepper motor driver is known for its ease of use, affordability, and versatility. The A4988 is specifically designed for driving bipolar stepper motors, which have two coils. It can control the current flow through the two motor coils to achieve controlled motion, rotation, and positioning of the motor shaft. Then, it supports microstepping, which allows finer control over the motor's movement by dividing each full step into smaller microsteps. Common microstepping modes include full-step, half-step, 1/4-step, 1/8-step, 1/16-step, and even 1/32-step. Microstepping enables smoother motion, reduces vibration, and provides higher precision. Then, A4988 features a straightforward step and direction interface, making it easy to control the motor's rotation and position. To make the motor move, a pulse (step signal) is applied to the "STEP" pin, while the "DIR" pin determines the direction of rotation based on the signal's logic level (high or low). Moreover, it allows users to set the maximum current delivered to the motor coils. This adjustable current limit helps prevent overheating and protects the motor from excessive current, optimizing motor performance and reliability. Lastly, A4988 includes built-in thermal shutdown and over-current protection features, which automatically disable the motor driver if the internal temperature exceeds a safe threshold or if the motor current exceeds the set limit.

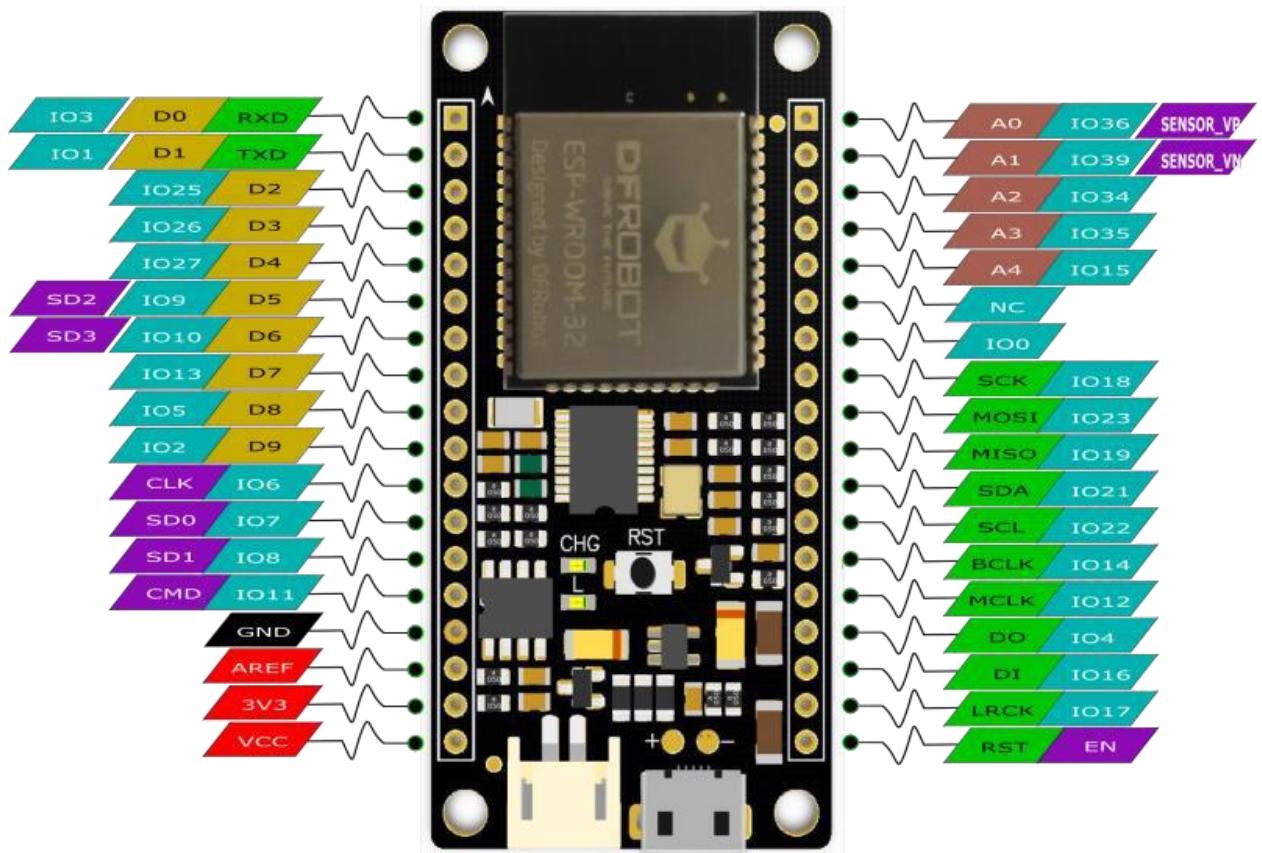


Figure 3.31: Firebeetle ESP32 module.

(Source: <https://botland.store/arduino-compatible-boards-dfrobot/9153-dfrobot-firebeetle-esp32-iot-wi-fi-bluetooth-5904422337612.html>)

Figure 3.31 shows the Firebeetle ESP32 module used in our project. The FireBeetle ESP32 is a development board based on the ESP32 system-on-chip (SoC) designed by DFRobot. It is a compact and feature-rich board that integrates the ESP32 microcontroller, offering a wide range of capabilities for various IoT (Internet of Things) and embedded projects.

The FireBeetle ESP32 is powered by the Espressif ESP32 SoC, which is a highly versatile and popular microcontroller known for its Wi-Fi and Bluetooth connectivity, as well as its dual-core processor. The ESP32 SoC on the FireBeetle board includes two Tensilica Xtensa LX6 microprocessor cores, which allow for multitasking and simultaneous handling of various tasks. Then, ESP32 chip on the FireBeetle board supports both Wi-Fi 802.11 b/g/n and Bluetooth 4.2/BLE (Bluetooth Low Energy) connectivity, making it suitable for a wide range of IoT applications and wireless communication. The FireBeetle ESP32 features an onboard PCB (Printed Circuit Board) antenna, providing convenience for wireless communication.

without the need for external antennas. The board has a micro USB port, which can be used for programming, debugging, and power supply. The FireBeetle ESP32 has a set of GPIO (General Purpose Input/Output) pins, allowing users to interface with various sensors, actuators, and external devices. The board includes a power management module that allows for low-power operation and efficient power management. The FireBeetle ESP32 can be programmed using the Arduino IDE, which is a popular and user-friendly development environment for creating Arduino-based projects. Then, it is part of DFRobot's FireBeetle series, which includes various compatible sensors, actuators, and accessories, making it easier to expand functionality and build complex projects. The FireBeetle ESP32 has a small form factor, making it suitable for space-constrained projects and prototyping.

Overall, the FireBeetle ESP32 is a versatile and easy-to-use development board that offers Wi-Fi and Bluetooth capabilities, making it ideal for IoT and wireless communication projects. Its compatibility with the Arduino IDE, which is suitable to use in our project to wirelessly control the stepper motor to control the steering system of the platform. Below are the pins in ESP32 module and their functions.

Table 3.7: All the pins and their functions in ESP32.

Digital GPIO Pins	The ESP32 features a set of GPIO (General Purpose Input/Output) pins that can be configured as either digital input or output. These pins can be used for connecting external sensors, actuators, or other digital devices.
Analog Input Pins	The ESP32 has several analog input pins that can be used to read analog voltage levels from external sensors or other analog devices.
UART Pins	The ESP32 has multiple UART (Universal Asynchronous Receiver/Transmitter) pins, which allow serial communication with other devices such as computers, microcontrollers, or peripherals.
I2C Pins	The ESP32 supports I2C (Inter-Integrated Circuit) communication and has dedicated pins for connecting I2C devices like sensors and displays.
SPI Pins	The ESP32 features SPI (Serial Peripheral Interface) pins for high-speed serial communication with other devices like external flash memory, SD cards, and other peripherals.
PWM Pins	The ESP32 provides several Pulse Width Modulation (PWM) pins that allow you to generate analog-like output signals for controlling motors, dimming LEDs, and other applications.

DAC Pins	The ESP32 has built-in DAC (Digital-to-Analog Converter) pins that can produce true analog voltage outputs.
Touch Pins	Some GPIO pins on the ESP32 can also be used as touch-sensitive pins, allowing you to detect touch or capacitive inputs.
Boot Mode Selection Pins	The ESP32 includes boot mode selection pins that determine the boot mode at power-up or reset, such as entering bootloader mode for programming or normal boot mode for running the user application.
EN Pin (Enable Pin)	The EN pin is used to enable or disable the ESP32 module.
USB Interface	The ESP32 has a USB interface, which can be used for programming, debugging, and serial communication.
External Interrupt Pins	The ESP32 has pins that can be used to trigger external interrupts, allowing you to respond to external events quickly.



Figure 3.32: Joystick module.

Figure 3.32 shows the joystick module which was developed by DFrobot. The Joystick Module from DFRobot is a popular input device designed to provide analog control for various electronic projects. It consists of a two-axis joystick that can detect movement along both the X and Y axes, as well as a push-button switch. The module is commonly used in robotics,

gaming, remote control applications, and other projects where precise and versatile control is needed. Here are the main features and components of the Joystick Module from DFRobot:

- Two-Axis Joystick: The primary component of the module is a two-axis joystick, allowing for analog control in two dimensions. The joystick can be moved in the X and Y directions, providing smooth and continuous analog input.
- X and Y Analog Outputs: The module typically provides two analog output signals corresponding to the X and Y movements of the joystick. These outputs can be connected to analog input pins of microcontrollers or other devices for reading the joystick position.
- Push-Button Switch: The joystick module includes a built-in push-button switch, which can be pressed by pressing down on the joystick. This push-button switch is useful for triggering specific actions or functions in your projects.
- Voltage Output: The X and Y outputs of the joystick provide analog voltage signals proportional to the joystick's position along the respective axes. The voltage range depends on the supply voltage provided to the module.
- Easy Integration: The Joystick Module is typically designed to be easily integrated into various electronic projects. It often comes with standard 3-pin headers or connectors for easy connection to microcontrollers and other devices.
- Durable Construction: The module is usually housed in a sturdy and compact casing, making it durable and suitable for use in various environments.
- Versatility: The Joystick Module can be used with a wide range of microcontrollers, development boards, and platforms, making it versatile for different applications.
- Input for Control Systems: The module's analog outputs are well-suited for controlling motor speeds, servo positions, or other parameters in control systems based on the joystick's position.

In our project, this joystick module is used to control the stepper motor on the platform, wirelessly, and thus can control the steering system to turn the platform into desired location.



Figure 3.33: Circuit connection of the controller (steering).

From the Figure 3.33, the wiring connection of the controller system for controlling the stepper motor (steering system) can be observed. However, the schematic diagram is not included due to the unavailability of this specific 3-pin joystick module in any electrical schematic diagram software.

The ESP32 module here is acted as a transmitter, in which it sends analog signal to the ESP32 module at the platform side, is connected to a 5V cell, and a 2-pin rocker switch is employed to interrupt or establish the voltage supply. The analog input of the joystick is connected to the A0 pin of the ESP32 module. For controlling the stepper motor, only the Xaxis of the joystick is utilized, enabling precise control over the direction and movement of the stepper motor.

This setup allows for straightforward and intuitive steering control, where the user can manipulate the joystick along the X-axis to determine the desired direction and speed of the stepper motor. By connecting the joystick's analog input to the A0 pin of the ESP32 module, the system can accurately interpret the joystick's position and translate it into corresponding movements of the stepper motor.

The stepper motor with the linear actuator was controlled using the joystick module from DFRobot, utilizing only the X-axis for direction and rotation control. The integration

involved connecting the joystick module's X-axis analog output to the appropriate input pin on the stepper motor driver, which facilitated the control of the stepper motor's movement.

By interpreting the analog voltage signal from the X-axis of the joystick module, the direction and speed of the stepper motor were precisely determined, enabling seamless control over the linear actuator's motion. The joystick's X-axis provided continuous analog input, allowing for smooth and responsive adjustments to the stepper motor's rotation.

In this setup, the Y-axis of the joystick module was not utilized, simplifying the control system and focusing solely on the X-axis for directional control. When the user manipulated the joystick along the X-axis, the stepper motor responded accordingly, guiding the linear actuator's movement based on the joystick's position.

This integrated configuration of the joystick module with the stepper motor and linear actuator offered precise and straightforward control over the linear motion, delivering an intuitive and efficient solution for steering system of the platform.

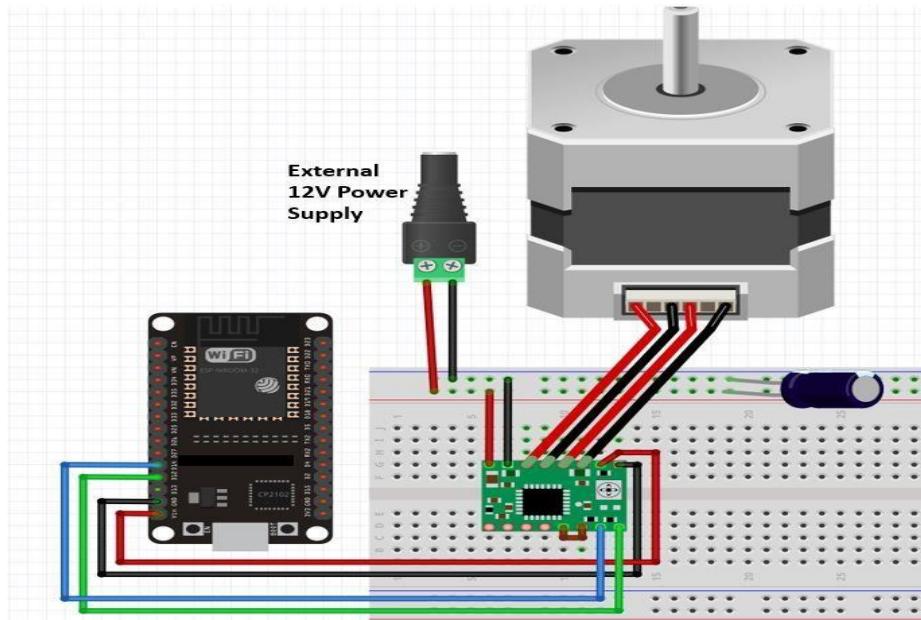


Figure 3.34: Schematic diagram of connection of stepper motor.

From the provided schematic diagram, a clear connection can be observed between the stepper motor and the A4988 driver, which is further connected to the ESP32 module. The ESP32 module here acts as a receiver, in which it receives the signal from the ESP32 module on the controller side. The stepper motor is powered by a 12V cell, and a 47-microfarad capacitor is

added to the 12V power line to prevent unstable voltage supply that could potentially damage the components.

For precise control over the stepper motor, the direction pin and step pin of the A4988 driver are connected to pin 3 and pin 1 on the ESP32 module, respectively. This arrangement allows the ESP32 module to exert control over the step and direction of the stepper motor's movements.

The 12V power supply efficiently energizes the stepper motor, enabling its reliable operation. The inclusion of a 47-microfarad capacitor ensures that the voltage supply remains stable, safeguarding the connected components from potential damage caused by fluctuations in the power source.

By connecting the direction and step pins of the A4988 driver to specific pins on the ESP32 module, precise and programmable control over the stepper motor's motion is achieved. This configuration allows the ESP32 module to regulate the direction and step signals, enabling smooth and accurate movement of the stepper motor.

The above setup and connections create a well-organized and functional system, providing a solid foundation for controlling the stepper motor with the ESP32 module and the A4988 driver. This implementation ensures stable and efficient operation, making it suitable for various applications that require precise motor control.

```

1 //include <esp_now.h>
2 #include <WiFi.h>
3
4 #define JoyStick_X A0
5
6
7 uint8_t broadcastAddress[] = {0x10, 0x97, 0xBD, 0x19, 0x62, 0xBC};
8
9 typedef struct struct_message {
10
11     float JS_Reading;
12
13 } struct_message;
14
15 struct_message myData;
16
17 esp_now_peer_info_t peerInfo;
18
19 // callback when data is sent
20 void OnDataSent(const uint8_t *mac_addr, esp_now_send_status_t status) {
21     Serial.print("\r\nLast Packet Send Status:\t");
22     Serial.println(status == ESP_NOW_SEND_SUCCESS ? "Delivery Success" : "Delivery Fail");
23 }
24
25 void setup() {
26
27     // Init Serial Monitor
28     Serial.begin(115200);
29
30     // Set device as a Wi-Fi Station
31     WiFi.mode(WIFI_STA);
32
33     // Init ESP-NOW
34     if (esp_now_init() != ESP_OK) {
35         Serial.println("Error initializing ESP-NOW");
36         return;
37     }
38
39     // Once ESPNow is successfully Init, we will register for Send CB to
40     // get the status of Trasmitted packet
41     esp_now_register_send_cb(OnDataSent);
42
43     // Register peer
44     memcpy(peerInfo.peer_addr, broadcastAddress, 6);
45     peerInfo.channel = 0;
46     peerInfo.encrypt = false;
47
48     // Add peer
49     if (esp_now_add_peer(&peerInfo) != ESP_OK){
50         Serial.println("Failed to add peer");
51         return;
52     }
53
54 }
55
56 void loop() {
57     // put your main code here, to run repeatedly:
58
59     float JS;
60     JS=analogRead(JoyStick_X);
61     myData.JS_Reading = JS;
62     Serial.println(myData.JS_Reading);
63     //Serial.println(x);*/
64
65
66     // Send message via ESP-NOW
67     esp_err_t result = esp_now_send(broadcastAddress, (uint8_t *) &myData, sizeof(myData));
68
69     if (result == ESP_OK) {
70         Serial.println("Sent With success");
71     }
72     else {
73         Serial.println("Error sending the data");
74     }
75     //delay(100);
76
77 }
78 }
```

Figure 3.35: Coding for Transmitter.

The provided code is an Arduino sketch written to control a joystick module and transmit its readings using ESP-NOW communication protocol. ESP-NOW is a communication protocol for fast and efficient data transmission between ESP8266 and ESP32 devices. In this code, an ESP32 board is used to read the analog input from a joystick's X-axis and send this data to a peer device with the specified MAC address using ESP-NOW.

- `#include <esp_now.h>` and `#include <WiFi.h>`: These are library inclusions necessary for using the ESP-NOW and Wi-Fi functionalities in the code.
- `#define JoyStick_X A0`: This line defines the analog input pin (A0) to which the X-axis of the joystick module is connected.
- `uint8_t broadcastAddress[]`: This array defines the MAC address of the peer device that will receive the joystick data.
- `typedef struct struct_message`: This line defines a custom data structure called `struct_message` that contains a single member `JS_Reading`, representing the joystick reading.
- `struct_message myData`: This line declares a variable `myData` of type `struct_message` to hold the joystick reading.
- `esp_now_peer_info_t peerInfo`: This line declares a variable `peerInfo` of type `esp_now_peer_info_t`, which will hold information about the peer device.
- `void OnDataSent(const uint8_t *mac_addr, esp_now_send_status_t status)`: This is a callback function that will be executed when data is sent via ESP-NOW. It prints the status of the sent packet (whether it was delivered successfully or not).
- `void setup()`: This function runs once when the ESP32 board is powered up or reset. It initializes the serial monitor, sets the ESP32 as a Wi-Fi Station, initializes ESP-NOW, and adds a peer for communication.
- `void loop()`: This function runs repeatedly after the setup. It reads the analog input from the X-axis of the joystick module using `analogRead(JoyStick_X)`, stores it in the `myData.JS_Reading`, and then sends this data to the peer device using `esp_now_send()`. The result of the sending process is printed on the serial monitor to indicate whether it was successful or not.

Please note that for this code to work, two ESP32 boards are needed. One board runs this code to send the joystick readings, and the other board runs a complementary code to receive and process the data. The receiving board should have the corresponding MAC address

specified in broadcastAddress[]. Additionally, both boards should be in range and have ESPNOW communication enabled for successful data transmission.

```
1 #include <esp_now.h>
2 #include <WiFi.h>
3
4
5 #include <Stepper.h>
6
7 const int stepsPerRevolution = 140;
8 const int stepPin = 1;
9 const int dirPin = 3;
10 int mappedValue;
11
12 const int maxSteps = 250;
13 const int minSteps = -250;
14 Stepper myStepper(stepsPerRevolution, stepPin, dirPin);
15
16 int js_val;
17
18 // Structure example to receive data
19 // Must match the sender structure
20 √ typedef struct struct_message {
21     float JS_Reading;
22 } struct_message;
23
24 struct_message myData;
25
26
27
28 √ void OnDataRecv(const uint8_t * mac, const uint8_t *incomingData, int len) {
29     memcpy(&myData, incomingData, sizeof(myData));
30
31     js_val = myData.JS_Reading;
32     Serial.print("JS Reading: ");
33     Serial.println(js_val);
34     Serial.println();
35
36 }
37
38 √ void setup() {
39
40     // Initialize Serial Monitor
41     //Serial.begin(115200);
42     myStepper.setSpeed(1000);
43
44
45     // Set device as a Wi-Fi Station
46     WiFi.mode(WIFI_STA);
47
48
49     // Init ESP-NOW
50 √ if (esp_now_init() != ESP_OK) {
51         Serial.println("Error initializing ESP-NOW");
52         return;
53     }
54
55     // Once ESPNow is successfully Init, we will register for recv CB to
56     // get recv packer info
57     esp_now_register_recv_cb(OnDataRecv);
58
59 }
60
```

```

61 void loop() {
62
63     int Value = myData.JS_Reading;
64     //int val = js_val;
65     mappedValue = map(Value, 0, 4095, 0, 1000);
66     //Serial.println(mappedValue);
67
68     static int stepCount = 0;
69     if (mappedValue>700){
70         // Check if the maximum step count is reached
71         if (stepCount < maxSteps) {
72             myStepper.step(10);
73             stepCount++;
74         }
75     }
76     else if (mappedValue<300){
77         // Check if the minimum step count is reached
78         if (stepCount > minSteps) {
79             myStepper.step(-10);
80             stepCount--;
81         }
82     }
83
84     else {
85         while (stepCount > 0){
86             myStepper.step(-10);
87             stepCount--;
88         }
89         while (stepCount <0){
90             myStepper.step(10);
91             stepCount++;
92         }
93     }
94 }
95 }
```

Figure 3.36: Coding for Receiver.

The provided code is an Arduino sketch that demonstrates how to control a stepper motor using the ESP-NOW communication protocol and receive data from another device. In this code, an ESP32 board is used to control the stepper motor's rotation based on the data received from another device, which is the joystick module (sender device) from the previous code.

#include <esp_now.h> and #include <WiFi.h>: These are library inclusions necessary for using the ESP-NOW and Wi-Fi functionalities in the code.

- #include <Stepper.h>: This library is used to control the stepper motor.
- const int stepsPerRevolution = 140;, const int stepPin = 1;, const int dirPin = 3;: These lines define the parameters for controlling the stepper motor, including the number of steps per revolution and the pins connected to the stepper motor driver for step and direction control.
- const int maxSteps = 250;, const int minSteps = -250;: These lines define the maximum and minimum steps the stepper motor can rotate in either direction.
- Stepper myStepper(stepsPerRevolution, stepPin, dirPin);: This line creates an instance of the Stepper class to control the stepper motor.
- int js_val;: This variable is used to store the joystick reading received via ESP-NOW.

- `typedef struct struct_message { float JS_Reading; } struct_message;`: This defines a custom data structure named `struct_message` that contains a single member `JS_Reading` representing the joystick reading.
- `struct_message myData;`: This line declares a variable `myData` of type `struct_message` to store the data received via ESP-NOW.
- `void OnDataRecv(const uint8_t *mac, const uint8_t *incomingData, int len)`: This function is a callback that is executed when data is received via ESP-NOW. It copies the received data into the `myData` variable and extracts the joystick reading value, storing it in `js_val`.
- `void setup()`: This function runs once when the ESP32 board is powered up or reset. It initializes the stepper motor, sets the ESP32 as a Wi-Fi Station, initializes ESP-NOW, and registers the callback function `OnDataRecv` to handle received data.
- `void loop()`: This function runs repeatedly after the setup. It reads the joystick reading value from `myData.JS_Reading` and maps it to a range between 0 and 1000 (`mappedValue`). Based on the `mappedValue`, the code controls the stepper motor's rotation. If the `mappedValue` is greater than 700, the stepper motor rotates forward (positive direction) in steps of 10 until the `maxSteps` value is reached. If the `mappedValue` is less than 300, the stepper motor rotates backward (negative direction) in steps of 10 until the `minSteps` value is reached. If the `mappedValue` is between 300 and 700, the stepper motor stops, and its position is held.

3.7 Controller Design

The controller has been designed to integrate both of the motor and steering control into one hardware device. However, both controls work separately since the ESC of the brushless motor driver come with embedded coding program that cannot be interfered with. Its also worth noting that some solutions have been discussed but limited resources and low practicality of the system planned make us put a stop to it and focusing on only physically combine the system. This problem also prevent us from including a preset route programs into the system as we could not acces the ESC coding.

With that bear in minds, the design of the controller has been modified to better suits the end product of the wiring system. Observing the Figure 3.37 below, the changes that have been made from the initial design are obviously the removal of the preset button, as well as the change of joystick type used from the stick type to analog type for ergonomic purposes. The

right side of the controller is designed purely based from the ESC controller to make the integration easier.

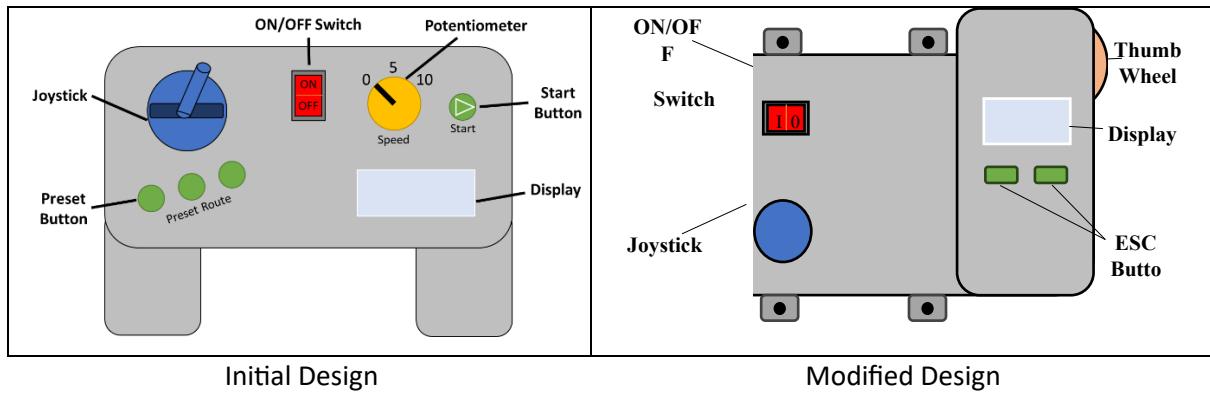


Figure 3.37: The initial (left) and modified (right) version of the controller design.

Upon finalizing the design of the controller, we then designed the 3D model using SolidWorks and make it into two parts; the top and the bottom parts of the controller. Based on the wiring connection that have been discussed previously, the controller will consist of three main boards which are the ESC board, the ESP32 board and the joystick module with addition of other components such as the power button, battery and thumb wheel. Both the ESC and the joystick module are been fixed at the top part as we need to interact with them, while the ESP32 is located at the bottom part to avoid any disturbance on the wiring connection. The 3D model can be observed below:

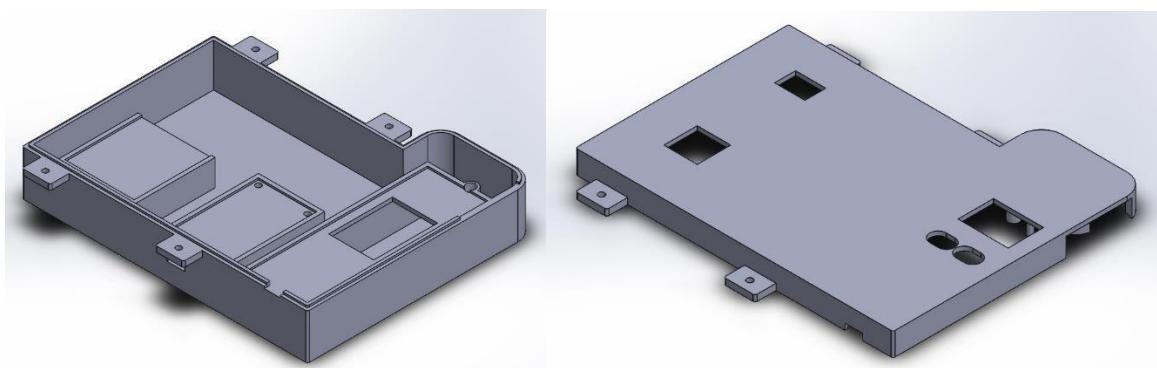


Figure 3.38: The 3D model of the top and bottom part of the controller.

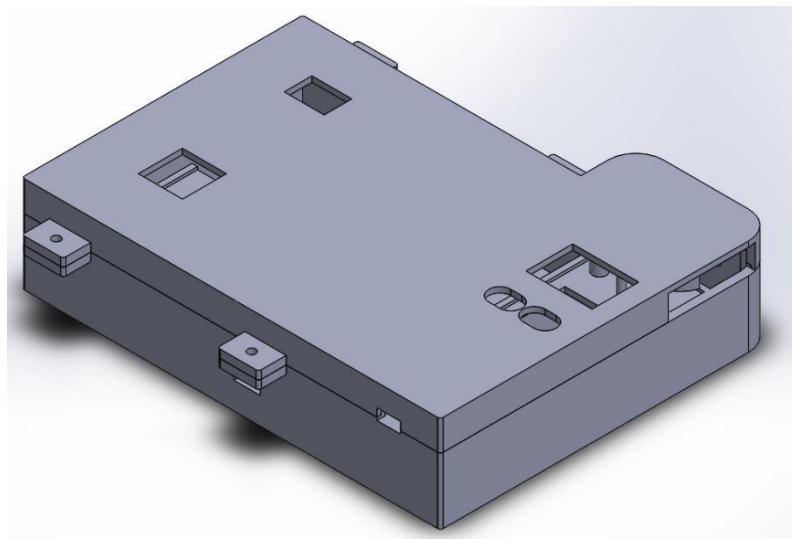


Figure 3.39: The assembly of the 3D model of the controller.

At this stage, we carefully crafted the design, especially on the ESC part of the design to ensure it closely resembled the original product while incorporating improvements and modifications to meet specific requirements. Once the 3D model was complete, we proceeded to the prototyping stage using Artillery Sidewinder X2 3D printer provided by the faculty to be utilized. By selecting PLA as the material, we successfully transformed the virtual design into a tangible prototype. This iterative process allowed for thorough testing and evaluation of the product's functionality, form, and fit, making it easier to identify any potential design flaws or areas for enhancement. Ultimately, this combination of cutting-edge software and advanced manufacturing techniques proved to be an efficient and effective means of developing and refining a product design.

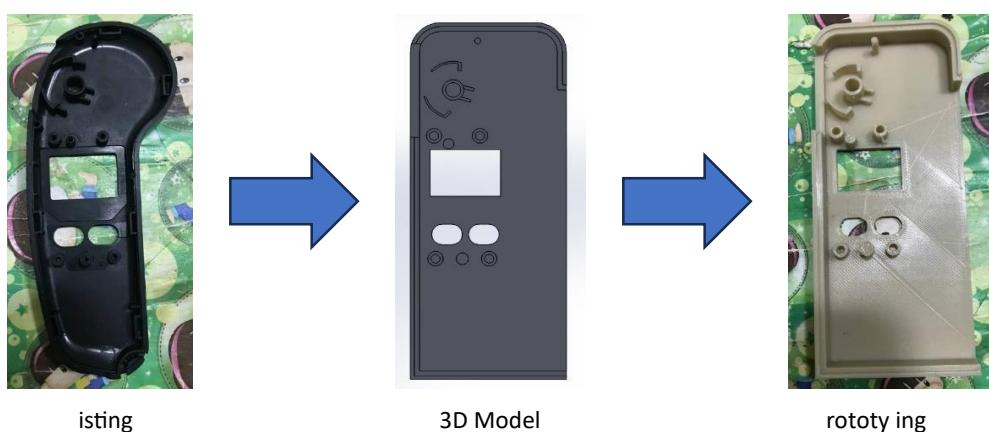


Figure 3.40: The prototyping process of the model design.



Figure 3.41: The Artillery Sidewinder X2 3D printer.

After the successful prototyping phase, the next step was to produce the final product. To achieve this, we carefully reviewed the feedback and results from the prototype testing and made necessary adjustments to the design. Once the final design was approved, we prepared the 3D model file for production and printed using the same 3D printer. Throughout the printing process, we closely monitored the production to detect any potential issues and make real-time adjustments as needed. After the 3D printing was completed, the product is assembled and connected with all the components required. The result was a meticulously crafted, production-ready item that successfully transformed the initial concept into a usable, practical product.

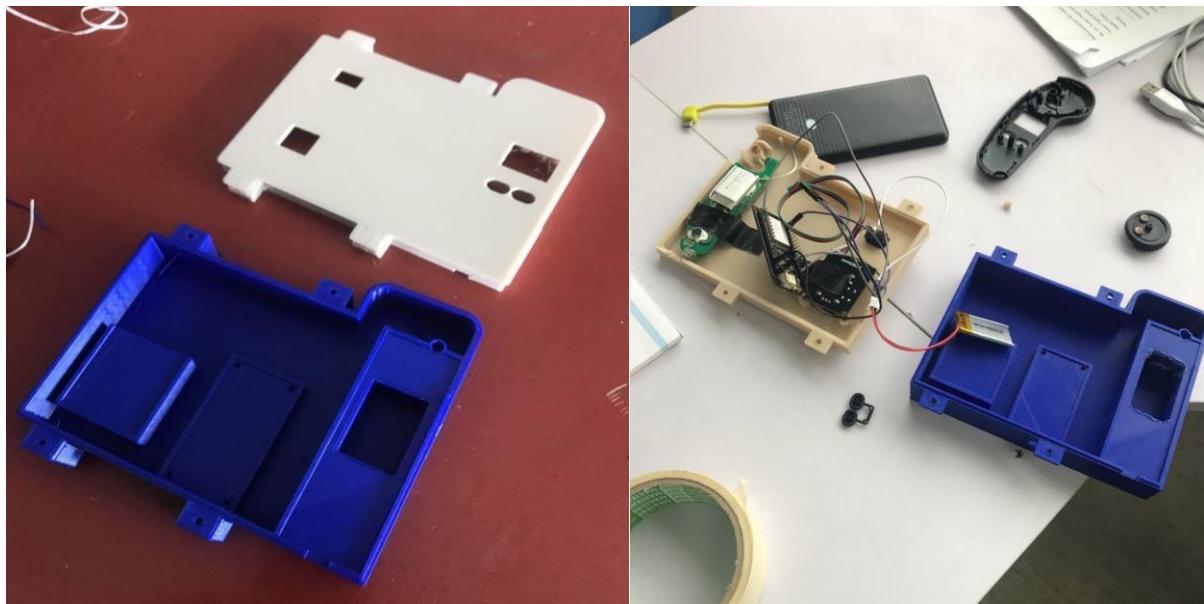
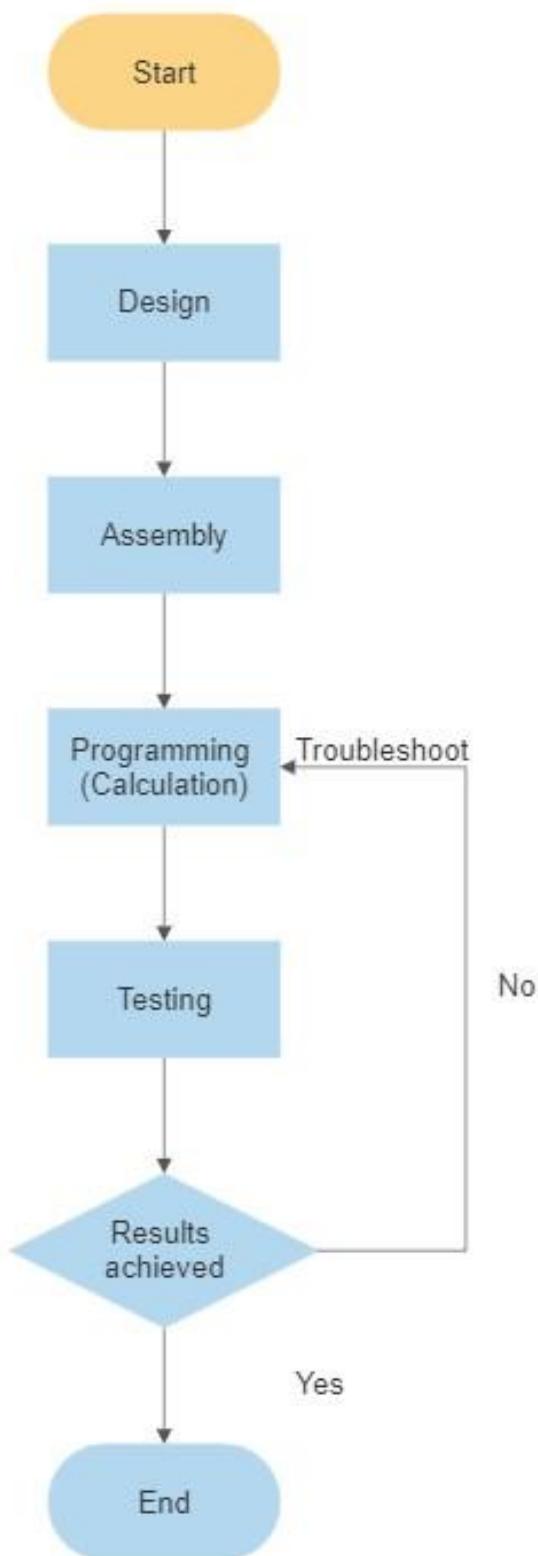


Figure 3.42: The assembly process of the controller.



Figure 3.43: The final product of the controller design.

4.0 EXPERIMENTAL PROCEDURE



4.1 PROCEDURE

4.1.1 Development and Testing

We are to design a steering system for the platform and to integrate a remote controller into the platform which controls the steering system and driving motors. The 3D modelling of the platform is done by using the 3D CAD Software SOLIDWORKS and the wiring diagram by using ...

In the assembly phase, our components are assembled onto the platform based on our design and also assembled into a remote controller. This includes joining the parts by screws or welding and wiring the steering system, motors and remote controller.

After the assembly is completed, we will proceed to program the microcontroller to control the steering and motors remotely while also including a preset route where the platform will travel according to the set route. In this phase, we will do the calculations for the steering angles and also the calculations for the preset route. The calculations are then programmed into the microcontroller.

During our testing phase, we will test programming so that the remote controllers accurately control the platform and also test the accuracy of our preset route by comparing our tests to a manual controlling of the platform through a series of routes by incorporating an accelerometer into our platform and obtaining the acceleration graphs of each test and comparing them.

After acquiring the results, we will decide whether to continue development after the results are achieved or to do troubleshooting when the results are unsatisfactory or incomplete. Troubleshooting includes recalculation and reprogramming of the microcontroller.

4.1.2 Assembly of the platform

For our platform, a metal frame and 2 motors with some wheels are already prepared for us. We will have to assemble the steering system and motors onto the metal frame while taking into account the specification requirements for the platform. Microcontrollers are installed which controls the steering and motors. A receiver is connected to the microcontroller for remote controlling.

4.1.3 Assembly of the remote controller

For the remote controller, it is expected that we design and 3D print a case where the components are assembled onto. A transmitter, a joystick and some buttons are connected to the microcontroller to remotely control the platform.

4.2 EXPECTED RESULTS

The main objective of this project is to create a VRU soft target platform that is portable and robust, easy to operate, low maintenance and long hours of use for the usage by students and lecturers for learning purposes and experiments.

Our VRU soft target platform should be portable and robust so that it is easy to carry and do tests with it while also being strong enough to withstand some force on it. Hence, a proper design and material selection is crucial in developing the platform.

It should also be easy to operate and low maintenance so that it is user-friendly. The remote controller is easy to use and intuitive. The platform also does not need much maintenance such as lubrication and also is not expensive to replace parts in the case of damaged components.

Furthermore, the platform should be able to be used for long hours before requiring recharging so that it can undergo many tests before the battery is low. With longer battery life, students and lecturers are able to use it to run tests with it better.

4.2.1 Results and Evaluation

After assembling all the parts of the platform and the controller, the measurements were carried out. It is important to note that the test was performed on normal road conditions. For the measurements in the evaluation, each test was performed three times and was recorded. The deviation of each test was saved. The deviation has been averaged over the three tests, so that the result is more accurate.

4.2.2 Testing the deviation from the centre.

In this test, the measurements for deviation and time were carried out in three different speeds (5km/h, 10km/h and 15km/h). For each speed, the results were averaged over 3 tests. The test was carried out by placing a line on the path of travel and the platform is then moved straight following the line for 10 meters, the deviation is measured by the distance of the platform from the line at the 10th meter.



Figure 4.2.1: Deviation testing setup

	5km/h		10km/h		15km/h	
	Time (s)	Deviation (m)	Time (s)	Deviation (m)	Time (s)	Deviation (m)
	11.15	1.27	6.40	1.34	4.23	3.90
	15.00	1.12	5.98	2.25	4.71	5.32
	10.65	1.73	6.33	2.30	4.25	2.00
Average	12.27	1.37	6.24	1.96	4.40	3.74

Table 4.2.1: Measurement table for deviation

Table 4.2.1 shows the measurement of time taken for the platform to travel 10 meters and the deviation from the centre line at the end. From the results, the deviation can be seen to increase as the speed increases. One of the factors could be caused by the smooth and slippery tyres that rotate

in place when more power is applied. There might not be enough traction for the platform to travel in a straight line ideally. Another factor is that the motors might not be rotating at the same speed due to the wear and tear of the motors.

4.2.3 Average Acceleration

From the measurements above, we can calculate the average acceleration by calculating the hypotenuse with the formula $a^2 + b^2 = c^2$ to find the displacement. We can calculate the average acceleration with the formula:

$$a = \frac{2(d - ut)}{t^2}$$

5km/h test

Displacement, d

$$d = \sqrt{1.37^2 + 10^2} = 10.09m$$

Acceleration, a

$$a = \frac{2(10.09 - 0)}{12.27^2} = 0.134 \text{ m/s}^2$$

Percentage Difference from testing value over the maximum estimated value.

$$Diff\% = \frac{|(0.134 - 3)| \text{ m/s}^2}{3 \text{ m/s}^2} = 0.95 \%$$

10km/h test

Displacement, d

$$d = \sqrt{1.96^2 + 10^2} = 10.19m$$

Acceleration, a

$$a = \frac{2(10.19 - 0)}{6.24^2} = 0.523 \text{ m/s}^2$$

Percentage Difference from testing value over the maximum estimated value.

$$Diff\% = \frac{|(0.523 - 3)| \text{ m/s}^2}{3 \text{ m/s}^2} = 0.65 \%$$

15km/h test

Displacement, d

$$d = \sqrt{3.74^2 + 10^2} = 10.68m$$

Acceleration, a

$$a = \frac{2(10.68 - 0)}{4.40^2} = 1.103 \text{ m/s}^2$$

Percentage Difference from testing value over the estimated value.

$$a = \frac{|(1.103 - 3)| \text{ m/s}^2}{3 \text{ m/s}^2} = 0.63\%$$

4.2.4 Turning radius test

In our second test, the radius of the turn at maximum steering angle is measured. The platform is turned left to the maximum steering angle and then moved until it has turned a 180° angle. The diameter is then measured from the starting point to the ending point. The test was repeated for speeds 5km/h, 10km/h and 15km/h.

5km/h		10km/h		15km/h	
Radius (m)	2.05	Radius (m)	2.20	Radius (m)	2.18
	2.02		2.27		2.15
	2.08		2.22		2.10
Average (m)	2.05	Average (m)	2.23	Average (m)	2.14

Table 4.2.4.1: Measurement table for turning radius

From the Table 4.2.4.1 we can see that the turning radius stayed relatively similar throughout the different speeds. The results show that the turning radius up to speeds of 15km/h have less errors because they only deviate slightly from each other.

4.2.5 Single motor test (Left and Right)

Our last test was done to test whether the left and right motors were moving at the same speed. This test is done to justify the reason of the large deviation in our first test. In the test, one of the motors was disconnected and only one is moving. The platform is moved for 3 meters at 5km/h. The test is done for both sides of the motor.

Right motor		Left motor	
Deviation (m)	0.79	Deviation (m)	0.58
	0.44		0.68
	0.79		0.58
Average deviation (m)	0.67	Average Deviation (m)	0.61

Table 4.2.4.2: Measurement of deviation for single motor

From the Table 4.2.4.2, the measurement shows that the deviation from the centre for both sides of the motor are different. This can mean that the left and right motors do not move at the same speed. This can be one of the reasons why the platform has a large deviation when travelling in a straight line.

5.0 CONCLUSION

In conclusion, in this project, we will be able to design a VRU soft target platform to be used for experiments by students and lecturers. We have managed to overcome the problems while designing the platform and remote controller and we think we are able to successfully complete this project. Our supervisor has also been a big help in keeping us on track and helping us overcome some of our problems during our weekly meetings.

Although our platform is less refined compared to big manufacturers who have enough funds to create a more advanced system, considering our time frame and budget with a small team of students working on this project, we think we have done a good job. We believe there is more that can be done which will improve this platform such as antenna to improve the range for the remote controller, etc.

Lastly, this project is interesting and has taught us some valuable lessons in design and building.

References

- [1] National Safety Council (2020). Motor Vehicle Deaths Estimated to Have Dropped 2% in 2019. <https://www.nsc.org/road-safety/safety-topics/fatality-estimates>
- [2] D.E. Kieras, S. Bovair, ‘The role of a mental model in learning to operate a device’, Cognitive Science, 8 (1984), pp. 255-274
- [3] C.D. Wickens, K. Gempler, M.E. Morphew, ‘Workload and reliability of predictor displays in aircraft traffic avoidance’, Transportation Human Factors, 2 (2) (2000), pp. 99-126
- [4] C.L. Baldwin, J. May, R. Parasuraman, ‘Auditory forward collision warnings reduce crashes associated with task-induced fatigue in young and older drivers’, International Journal of Human Factors and Ergonomics, 3 (2) (2014), pp. 107-120
- [5] R. Spicer, A. Vahabaghaie, G. Bahouth, L. Drees, R. Martinez von Bülow, P. Baur, ‘Field effectiveness evaluation of advanced driver assistance systems’, Traffic Injury Prevention, 19 (sup2) (2018), pp. S91-S95
- [6] J.B. Cicchino, ‘Effectiveness of forward collision warning and autonomous emergency braking systems in reducing front-to-rear crash rates’, Accident Analysis & Prevention, 99 (Pt A) (2017), pp. 142-152, 10.1016/j.aap.2016.11.009
- [7] L. Yue, M. Abdel-Aty, Y. Wu, L. Wang, ‘Assessment of the safety benefits of vehicles’ advanced driver assistance, connectivity and low-level automation systems’, Accident Analysis and Prevention, 117 (2018), pp. 55-64, 10.1016/j.aap.2018.04.002
- [8] J.B. Cicchino, ‘Effects of blind spot monitoring systems on police-reported lane-change crashes’, Traffic Injury Prevention, 19 (6) (2018), pp. 615-622, 10.1080/15389588.2018.1476973
- [9] J.B. Cicchino, ‘Real-world effects of rear cross-traffic alert on police-reported backing crashes’, Accident Analysis & Prevention, 123 (2019), pp. 350-355, 10.1016/j.aap.2018.11.007

- [10] NTSB ‘The Automatic Emergency Braking (AEB) or Autopilot systems may not function as designed, increasing the risk of a crash’, USDOT, Washington, DC (2017)
- [11] L. Onnasch, C.D. Wickens, H. Li, D. Manzey, ‘Human performance consequences of stages and levels of automation: An integrated meta-analysis’, Human Factors, 56 (3) (2014), pp. 476-488, 10.1177/0018720813501549
- [12] C.A. DeGuzman, B. Donmez, ‘Knowledge of and trust in advanced driver assistance systems’, Accident Analysis and Prevention, 156 (2021), p. 106121, 10.1016/j.aap.2021.106121
- [13] ‘VRU Testing for ADAS and AV’, <https://www.messring.de/en/blog/vruteesting-for-adas-and-av/> (Retrieved from 13/04/2023)
- [14] WHO, ‘Global Status Report on Road Safety 2018’, World Health Organization (Ed.), Geneva (2018)
- [15] Olszewski P., Szagała P., Rabczenko D., Zielińska A., ‘Investigating safety of vulnerable road users in selected EU countries’, J. Saf. Res., 68 (2019), pp. 49-57
- [16] Oikawa S., Matsui Y., ‘Features of serious pedestrian injuries in vehicle-topedestrian accidents in Japan’, Int. J. Crashworthiness, 22 (2) (2017), pp. 202-213
- [17] Qiang C., Yong C., Bostrom O., Ma Y., Liu E., ‘A comparison study of car-topedestrian and car-to-e-bike accidents: Data source: The China in-depth accident study (CIDAS)’, SAE 2014 World Congress & Exhibition, SAE International, Detroit, MI, United States (2014)
- [18] ‘SPT VRU Platform’,
<https://www.abdynamics.com/en/products/tracktesting/adas-targets/soft-pedestrian-target> (Retrieved from 13/04/2023)
- [19] ‘About us, AB Dynamics’,
<https://www.abdynamics.com/en/about-us>
(Retrieved from 14/4/2023)

- [20] ‘LaunchPad VRU Platform Range’,
<https://www.abdynamics.com/en/products/track-testing/adas-targets/launch-pad>
 (Retrieved from 14/4/2023)
- [21] Richard Kurniawan , ‘Design and Implementation prototype of goods carrier robot using microcontroller’, Faculty of Information Technology, Ciputra University.
- [22] Mark W. Spong, Seth Hutchinson, and M. Vidyasagar, ‘Robot Modeling and Control – First Edition’, JOHN WILEY & SONS, INC. (2006)
- [23] ‘The Vector Field Histogram - Fast Obstacle Avoidance for Mobile Robots’, IEEE Journal on Robotics and Automation, 7(3), pp. 278-288.
- [24] Craig, “Introduction to Robotics: Mechanics and Control,” 3rd Edition, Pearson Prentice Hall, Upper Saddle River, 2005.
- [25] Čerkala, Jakub & Klein, Tomáš & Jadlovska, Anna. (2015). ‘Modeling and Control of Mobile Robot with Differential Chassis’, Electrical Engineering and Informatics 6 : proceedings of the Faculty of Electrical Engineering and Informatics of the Technical University of Košice. 651-656.
- [26] Halmetschlager-Funek, Georg & Prankl, Johann & Vincze, Markus. (2016).
 ‘Towards Agricultural Robotics for Organic Farming’.
- [27] Jayaraman, Kuberan & Reddy, Kandlagunta. (2019). ‘DESIGN AND FABRICATION OF STEERING SYSTEM USING SERVO MOTORS.’ 6. 266-270.
- [28] ‘Self-driving bicycle developed by huawei engineers can operate unmanned’ (2021),
<https://www.designboom.com/technology/self-driving-bicycle-huaweiengineers-operate-unmanned-06-14-2021/> (Retrieved from 14/4/2023)
- [29] “LINEAR ACTUATORS WITH STEPPER MOTORS”,
<https://en.nanotec.com/products/157-linear-actuator-stepper-motor>),
 Retrieved form 12/08/2032.
- [30] “Ball Screw Linear Slide Table Stage Actuator w/Stepper Motor Stroke 100mm/400mm”, (<https://www.ebay.com/item/173825867326>), Retrieved from 13/08/2023.

- [31] “Tech Explained: Ackermann Steering Geometry”, (<https://racecar-engineering.telegraph.co.uk/articles/tech-explained-ackermann-steering-geometry/>), Retrieved from 16/08/2023.
- [32] Murali, Vishnu & M R, Akash & Pandit, Suparshwa. (2020). Design and Development of Four-Wheel Steering for All Terrain Vehicle (A.T.V). 10.13140/RG.2.2.11838.92482.
- [33] Topaç, Mehmet & Deryal, Uğur & Bahar, Egemen & Yavuz, Guven. (2015). Optimal kinematic design of a multi-link steering system for a bus independent suspension: An application of Response Surface Methodology. Mechanika. 21. 10.5755/j01.mech.21.5.11964.

APPENDIX

GANTT CHART

