Mobile Controlled Robot(4WD/4WS) for agricultural use

Engineering Project Report

New Zealand Diploma in Engineering (Electrical)

Department of Electrotechnology

Unitec Institute of Technology

Project Supervisor: Deepinder Sidhu

Kyowon Park

1540216

2021

Declaration of Originality

This report is my own unaided work and was not copied from nor written in collaboration with any other person.

Name: Kyowon Park

Abstract

Growth in population has been a problem in farming industry for several years and the agricultural robots started developing around 1920s. The number of workers required on the field are becoming more insufficient around the world. There are many agricultural robots that can perform tasks such as weed spraying, seed planting and harvesting. My project aims to aid in agriculture by making a prototype robot that can provide a platform for various agrobot projects to be attached. The methodology approach was to research online to gather information on parts required and experimenting with the information. This robot is easy for people to use by just simply connecting with Bluetooth on smartphone. One custom built prototype PCB has all the modules that are needed for this robot which includes modules such as microcontroller, voltage regulators and motor driver. Advantage of this PCB is that it can have supply voltage up to 24V DC where the robot can be easily modified for more power, speed and longer lasting. The use of 4 servo motors for each wheel allows the robot to have independent steering and give great advantage of moving freely in small areas like green house. My robot aims for user to easily modify hardware and software and not only can this be used in agriculture, but it can provide wide range of application.

Acknowledgements

I would like to thank lecturer Deepiner Sidhu for providing, supervising and giving ideas for this project. I also would like to thank Dr Momen Bahadornejad for leading this paper. Special thanks to my friend for lending me his 3D printer for constructing my prototype. Thanks to my families for their support, encouragement and feedbacks for improvements. Finally, I would like to thank lecturer Anthony Lai and Dr Nigel Yee for attending to my presentation and demonstration session.

Table of Contents

List of Figures

Chapter One: Introduction

1.1 Background

Agricultural robot started developing around 1920 and automatic vehicle was implemented into agriculture between 1950s and 60s. In recent years, the population around the world has increased rapidly where the demand for crops or food grown in farms has increased. The problem is that the shortage of workers is increasing as time goes on.

1.2 Aims and Significance of the project

The aim of my project is to build a prototype smart phone controlled robot vehicle that will provide a platform for other projects such as LED weed detector for agricultural use. My project will have four individual steering mechanism which allows the robot to move freely in many directions without having to move its body. New Zealand farmers relies on quad bikes for their farm vehicles where 80 000 to 100 000 quad bikes are used in NZ. Despite the fast transportation and the ability to attach accessories like sprayer, the quad bike has many problems. The driver is required to wear safety outfit due to the harmful chemicals that herbicide contain, and the accidents caused by the quad bikes are significant where 346 quad bike accidents reported between 2012 to 2018.

Remote controlled vehicles can eliminate these problems and aid in the agriculture crisis that is currently happening. Robots do not get tired and can get a lot more jobs completed which can solve labouring issues.

1.3 Structure of the report

Introduction chapter has briefly articulated the background of the research and the aim and significance of the project. The literature review briefly explains the history and the current work of agricultural robot. The methodology chapter outlines the methods and processes used to conduct the project. The final chapter concludes the results of the project and provides recommendations for future studies.

Chapter Two: Literature Review

Robotics technology including the agricultural robot is advancing. Development of autonomous agricultural vehicles were developed in 1950s and it was not until the 1980s when the computer technology was implemented into the agricultural robot. MAR-1 Agricultural Robot was developed in the early 1980s at Moscow Institute of Agricultural Engineers (figure 1). MAR-1 was made for cattle-breeding industry and was developed so it can perform autonomous entering and exiting of the site.

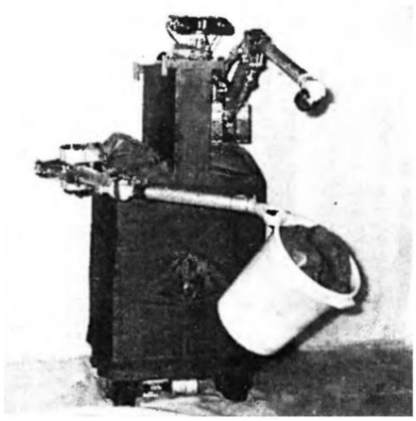


Figure 2.1 -first model of MAR-1 (Robot, 2012)

There are many types of agricultural robots in modern days equipped with advanced technology. However, there lots of future works for these robots for example, robot sensor cannot identify a certain type of plants or fruits for harvesting and fruits do not grow from the branches identically. Most of the agricultural robots are designed for a certain tasks and must be replaced or purchase another type of robot for different tasks.

In New Zealand, the Robotics Plus is currently developing Unmanned Ground Vehicle (figure 2.2) to be used in the orchard field. This project is collaborated with university of Auckland and Waikato. This vehicle aims to carry their various agricultural projects such as the harvester and navigate through the orchard autonomously. A disadvantage of this vehicle is that it only has two wheel steering system which can only be operated in the large open areas.

Figure 2.2- UGV by Robotics Plus (Harris, 2020)

Chapter Three: Methodology

This chapter will explain the methodology that was used to complete designing and constructing of prototype. This project was completed by using internet to gather information and carrying out experiments. There were three main steps which were planning, testing and evaluation.

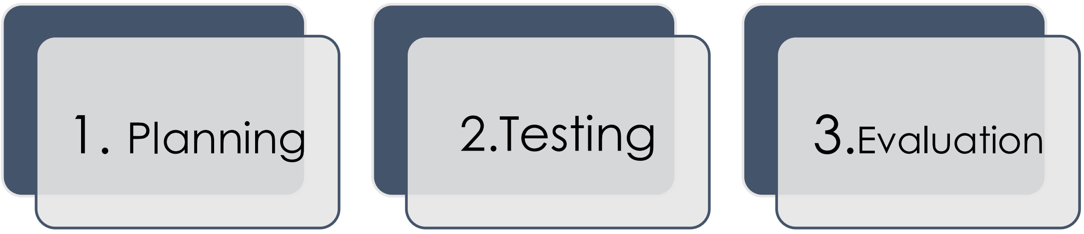


Figure 3-Project Steps

3.1 Planning

First step was to identify the main functions and objective for my project and then decide on the requirements for the hardware and software. There were three main objectives for the project which were: controlled using smart phone by Bluetooth connection, can steer in different modes and powered by batteries. Once the objectives were identified, I carried out research on mobile controlled robot online to gather information on the components and design ideas of the prototype. Figure below is the hardware design.

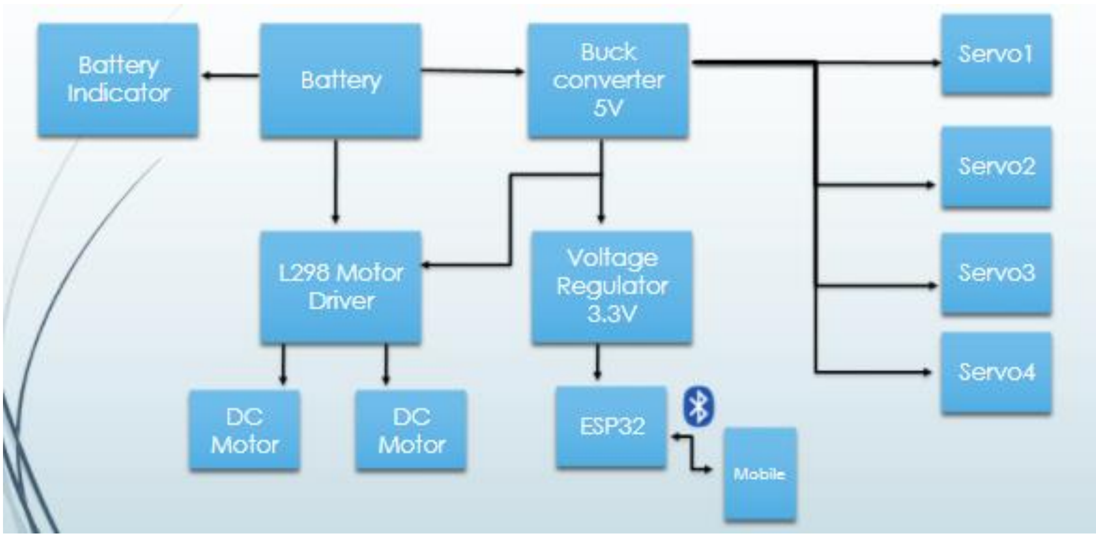


Figure 3.1-Hardware design

3.2 Hardware

The aim for selecting parts were to meet the budget of under $200 and be able to perform the main functions to meet the objectives. Below is the list of the hardware that was used to complete this project.

Part/component	Qty	Seller	Cost ($NZD)
ESP32	1	Trademe	13.99
L298 IC	1	Trademe	4.99
LM2596	1	DigiKey	6.12
LD1117 3.3V	1	DigiKey	0.70
Inductor 47uH 3A	1	DigiKey	2.46
Dc motor with wheels	4	Trademe	47.6
18650 battery holder	1	Trademe	3.99
PCB board 120mm x 80mm	1	Trademe	4.99
Pin header 1x3	4	Trademe	2.50
2 Pin Terminal Block(10pc)	1	Trademe	3.99
Jumper Wire pack	1	Trademe	8.99
330uf 35V	2	Surplustronics	0.70
MG995 servo motor	4	Amazon	28.09
MISC			30
			Total 158.41

Table 3 – Component list table

3.2.1 Main components

TT DC motor

This is a generic 3-6V DC motor for simple robotics and affordable. There was no need for more powerful motors for this prototype as its main objective can be made with TT DC motors.

Figure 3.2-TT DC motor

Battery
Two Li-ion rechargeable batteries were the main supply which is 8.4V when fully charged. The supply can be upgraded up to 24V if needed.

Figure 3.3-Li-ion 3.7V battery

LM2596

LM2596 is a simple switching power converter which can have an input up to 45V DC and step down to 5V DC with 3A output current. Supplying more than 24V is not recommended as the efficiency of the converter drops and will have heating problem. The model used for this prototype is the fixed 5V version and has the efficiency of 80% with 12V input. The 5v is required by the servo motors and the IC input of L298 motor driver.

Figure-3.4LM2596 5V

L298
The L298 is a dual h-bridge motor driver which can have maximum input voltage of 46V and has output current of 2A. It is recommended that diodes are used at the output

as this IC does not have internal protection for inductive load. This motor driver was used so this prototype can be upgraded to 12V DC motors.

Figure 3.5-L298HN motor driver

ESP 32

The ESP 32 has built in Bluetooth and Wi-Fi feature and is known for its low power consumption. ESP32 has the Bluetooth 4.2 and its range can be up to 50 meters outdoors and about 10 meters indoors. It is recommended to use 10uf and 0.1uf decoupling capacitor in parallel at the 3.3V pin to avoid brown out issues. ESP32 features 34 GPIO pins and 18 analog channels. This prototype will be using 10 analog channels.

Figure 3.6-ESP32 wroom 32

MG995 Servo motor

By using one servo motor for each wheel can allow the robot to have individual steering feature. MG995 is affordable and provide good amount of strength where each motor can lift around 9.4kg/cm when operating at 5V.

Figure 3.7-MG995 servo motor

3.3 Software

The required software to design the prototype including android app and PCB were OnShape, Arduino IDE, Easy EDA and MIT App inventor.

3.3.1 OnShape

OnShape is a free online software for 3D designing and it is very similar to Solid works. I used this software to design the body and wheel attachments. The requirement for the robot would be to have a ground clearance from the body of the robot and to have four legs to hold servos and the DC motors. There are four main parts that were designed using OnShape, base, arms, legs and holder. The base was a general flat platform for the PCB and battery to be placed on top. Figure below is the finished 3D design of the prototype robot.

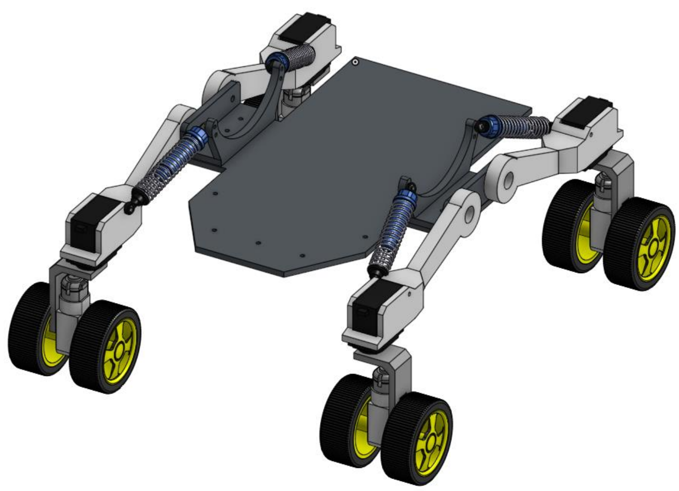


Figure 3.8- 3D design of the robot

3.3.1.1 Arms

Arms will hold the servos in place and there were errors during designing stage where the servos did not quite fit properly at the first attempt (figure 3.9) and adjustment had to be made.

Figure 3.9

Figure 3.10

Another issue of the design was that the arms holding the servos were unstable when turning and this was fixed by placing the shock absorber to hold the arms and the body together.

3.3.1.2 Legs

The leg is designed so that the DC motors can sit in upright position to give more height for the robot. The leg attaches to the servos, so it can be moved according to the servo angles.

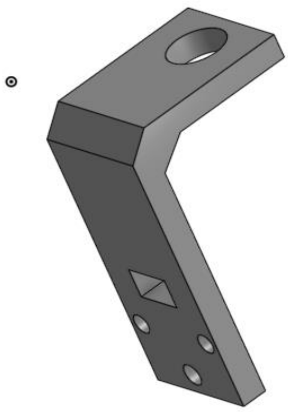


Figure 3.11-Designed leg

Figure 3.12-assembled leg

3.3.1.3 Holder

Due to the length of the arm, the arms were very unstable when turning, it would wobble around sideways without any support. The arm holder will hold the shock absorber and the arms together to give more stability when the robot is moving. M5 screws are used for the arms and the size of the shock absorber is 1/10 RC 90mm shock absorber. The extra holes on the horn looking design is for giving options to change the size of shock absorber.

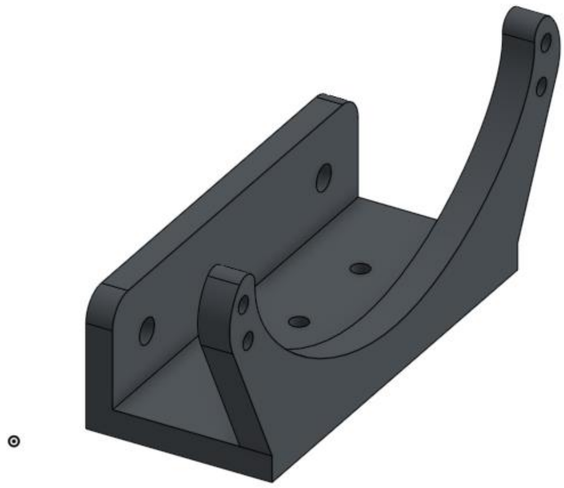


Figure 3.13-Designed holder

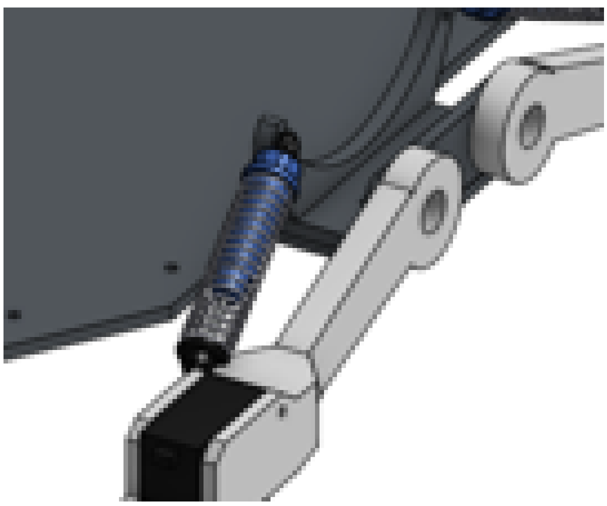


Figure 3.14-assembled holder

3.3.2 Arduino IDE

Arduino IDE was the software used for coding and programming the ESP32. First step was to define variables and attach motors to ESP32 output pins. Once the motors were attached to the pins, next step was to set up Bluetooth connection and analogue channels for all motors (figure 3.15).

```
void setup() {

  Serial.begin(115200);
  SerialBT.begin("ESP32_PARK"); //Bluetooth device name

  pinMode(in1, OUTPUT);
  pinMode(in2, OUTPUT);
  pinMode(in3, OUTPUT);
  pinMode(in4, OUTPUT);


ledcSetup(ServoChannel1, freqservo, resolution16);
ledcSetup(ServoChannel2, freqservo, resolution16);
ledcSetup(ServoChannel3, freqservo, resolution16);
ledcSetup(ServoChannel4, freqservo, resolution16);

  ledcAttachPin(servoPin,ServoChannel1);
  ledcAttachPin(servoPin1,ServoChannel2);
  ledcAttachPin(servoPin2,ServoChannel3);
  ledcAttachPin(servoPin3,ServoChannel4);

  ledcSetup(pwmChannel1, freq, resolution);
  ledcSetup(pwmChannel2, freq, resolution);
  ledcAttachPin(en1, pwmChannel1);
  ledcAttachPin(en2, pwmChannel2);


}
```

Figure 3.15-Setup loop

The resolution 16 bit was set up for the servo as it allows the servo to have more steps when moving from 0 to 180 angles. To find the normal angle range of 0 to 180, the conversion had to be made. The default pulse width range for MG995 servo motor is 500us (0 degree) to 2500us (180 degree). Using this formula, I was able to find new range**,**

min = (pulse width low)/(20ms/2^16)

The calculated ranges were from 1638 to 7864. However, these values are very difficult to identify angles of servos, so the Mapping function was used to convert these values in to 0 to 180 range shown in figure 3.16.

```
ledcWrite(2,map(angle,  0, 180, 1638, 7864));
ledcWrite(3,map(angle2, 0, 180, 1638, 7864));
ledcWrite(4,map(angle3, 0, 180, 1638, 7864));
ledcWrite(5,map(angle4, 0, 180, 1638, 7864));
```

Figure 3.16 - mapping of angle range

The main body (figure 3.17) of the code contains if else statements to call the functions of motors. when there is a Bluetooth data, it will call the corresponding function to move the motors as required.

```
    if(bt_data == 1){ forward();

    if(Speed<255) {              // line of code for gradually speeding up when going foward.
  ledcWrite(pwmChannel1, Speed);
  ledcWrite(pwmChannel2, Speed);
  Speed+=1;
}
    }

else if(bt_data == 2){backward();

    if(Speed<255) {             // line of code for gradually speeding up when going backward.
  ledcWrite(pwmChannel1, Speed);
  ledcWrite(pwmChannel2, Speed);
  Speed+=1;
}
}

else if(bt_data == 3 ){  turnLeft(); }
else if(bt_data == 4 ){  turnRight(); }
else if(bt_data == 5){  Stop(); Speed = 220;    }  // resets the speed to 220

   else if(bt_data == 6) { straight(); }    // servo controls
   else if(bt_data == 7) { circle();   }
   else if(bt_data == 8) { left();     }
   else if(bt_data == 9) { Hardleft();     }
   else if(bt_data == 10) { right();     }
   else if(bt_data == 11) { Hardright();     }
   else if(bt_data == 12) { horizontal(); }
   else if(bt_data == 13) { DiaLR(); }
   else if(bt_data == 14) { DiaRL(); }
  delay(20);
```

Figure 3.17 - Main body of code

When transferring code to the ESP32, it is important to choose the right serial port and board. In my case I had to choose the “fire-beetle ESP32” board. Serial port depends on the USB port the user chooses to connect.

3.3.3 Easy EDA

My goal was to design and assemble one PCB board half the size of an average smartphone which consists of all the modules that were required for the prototype robot. Before designing a schematic diagram, I used the data sheets of main components to select right capacitors and inductors according to my desired input and output voltages. The software I used to design schematic diagram was Easy EDA. Once the schematic was completed, I designed the layout of the PCB. The cost for ordering manufactured PCB was well over 100 NZD due to the SMD service and shipping. This was going over the budget, so I chose to solder and assemble using perfboard and through hole components.

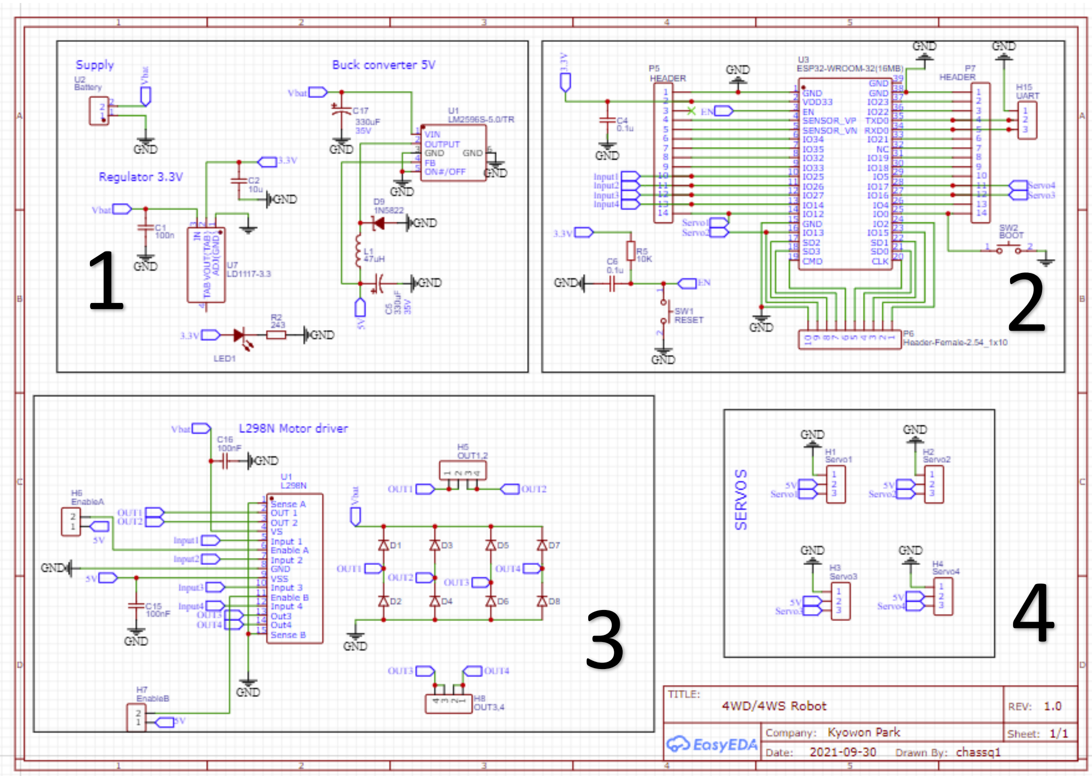


Figure 3.18 – Schematic Diagram designed by Easy EDA

Section 1 is the power supply modules which includes input terminal, voltage regulator and buck converter. The capacitor and inductor value for the LM2596 was chosen from the tables in data sheet for maximum 24V input. The LD1117 3.3V regulator is designed according to the typical application.

Section 2 is the ESP32 with the header pins. The ESP32 has certain pin connection to get in to transfer mode which requires to for enable pin and I0 pin to be high at the same time and enable pin low while the I0 is still high. This can be achieved by placing switch on these pins. There is header pin connected to the Tx, Rx and GND of the ESP32 for UART programmer.

Section 3 is the L298 motor driver. Stepped down 5V is going to the Vin of the IC and the supply voltage directly goes to Vs for the DC motors. Outputs and enable pins are connected accordingly to the selected ESP32 pins. 0.1uf capacitors are connected at both input pins of IC to smooth out the DC voltage.

Section 4 is the header pins for the four servo motors.

When placing the ESP32 chip on to a PCB, it is important for the antenna part of the ESP is not in contact with the copper layer. The layout design for the PCB is shown in figure 3.18 below.

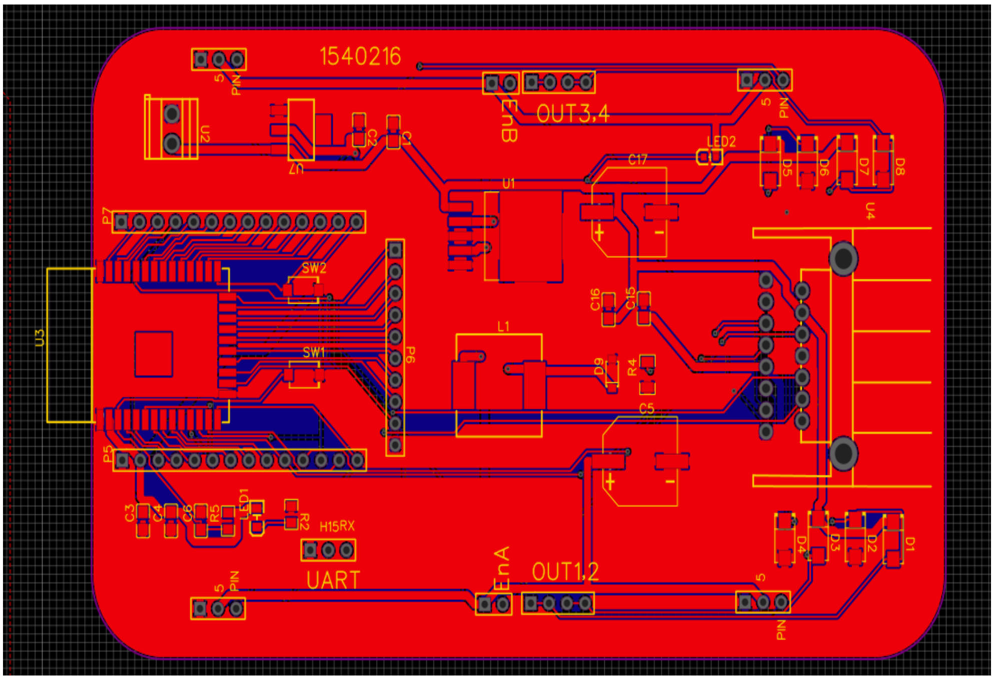


Figure 3.19 – Layout Design Topside

The PCB will have extra header pins around the ESP32, so it can add more servos and motors if it is required in the future. Figure 3.20 is the generated 3D view of my PCB.

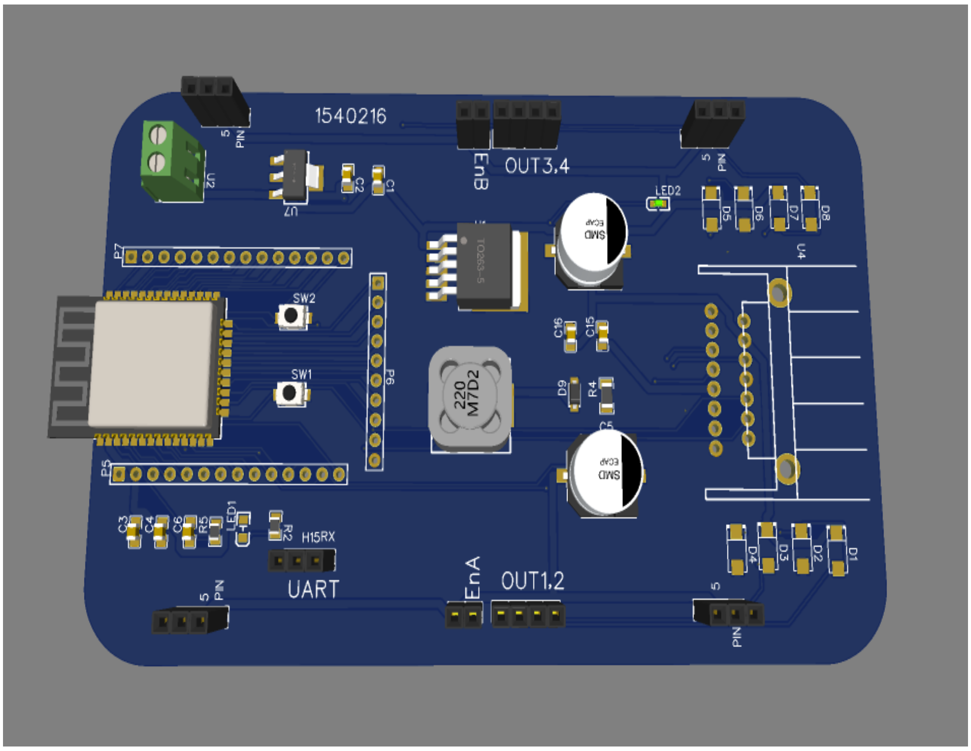


Figure 3.20 – 3D view of PCB design

3.3.4 MIT App Inventor

The fourth software required for this project was MIT App Inventor to create a controlling app for a smartphone. MIT App inventor is a free online visual app development environment currently provided by the Massachusetts Institute of Technology. The app is programmed using block coding system. The block code will send the Bluetooth data to the ESP32 and will be able to control the robot via smartphone.

Figure 3.21 – MIT App Inventor designing

Figure 3.22 shows the android application that was designed by MIT App Inventor. Touching the connect button will bring up the list of Bluetooth devices that are visible for connection. Once Bluetooth is connected the red "Disconnected" sign will turn to blue "Connected" sign. User can disconnect by touching the "Disconnect" button.

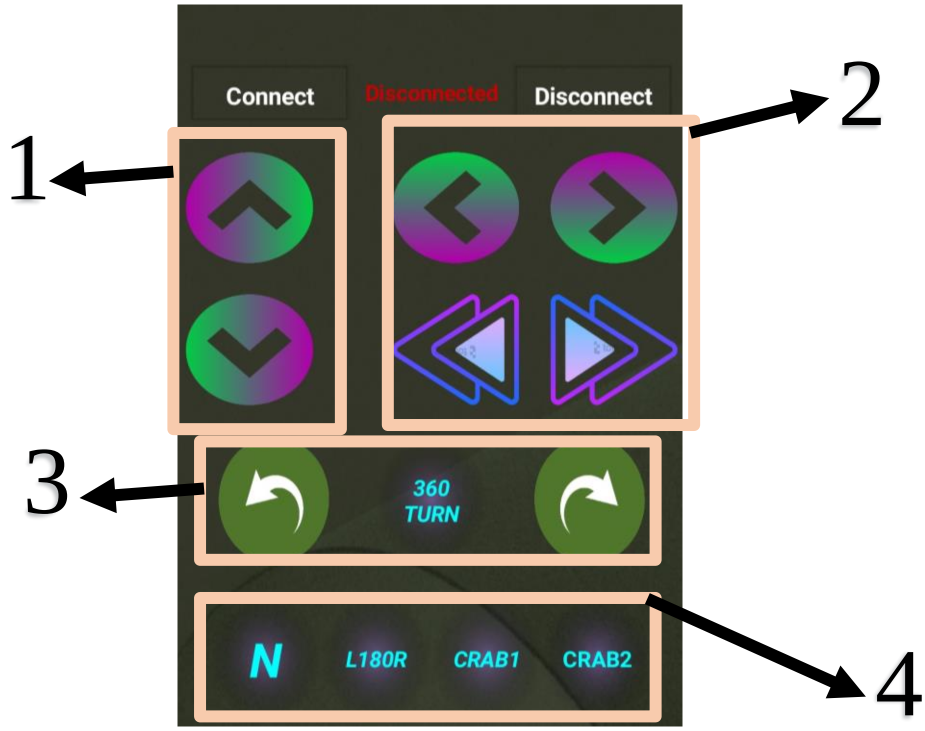


Figure 3.22 – Design of android App

Section 1 – these are the buttons that control the DC motors to move forward and backward directions.

Section 2 – the top two buttons have one arrow for left and right and this is the front two wheels steering. The bottom two buttons have double arrow, and this is for sharp turning by including use of rear wheel steering.

Section 3 – this section is for 360 mode. When the 360Turn button is pressed, the servos will move to the set position and by using the two buttons on each side of the 360 buttons, user can rotate the robot clock wise and anti-clock wise.

Section 4 – there are four more steering modes which includes N-normal position, L180R-position servos to 180 degree, Crab1-position servos diagonally from right to left and Crab2-position servos diagonally from left to right.

3.4 Testing

3.4.1 Simple Code and App

I started with simple code and simple app design to test my Bluetooth control with Arduino Uno. There were simple example codes in Arduino IDE for LED blinking. I tried to modify this code to Bluetooth control. After successfully getting the Bluetooth control to work, I got myself familiarized with coding for the servos. During this testing stage, I was able to get the ideas of how the programming was going to be implemented to my prototype.

The testing of simple Bluetooth control using smartphone shown in figure 3.23 and figure 3.24 below.

Figure 3.23 – Simple LED control ON and OFF

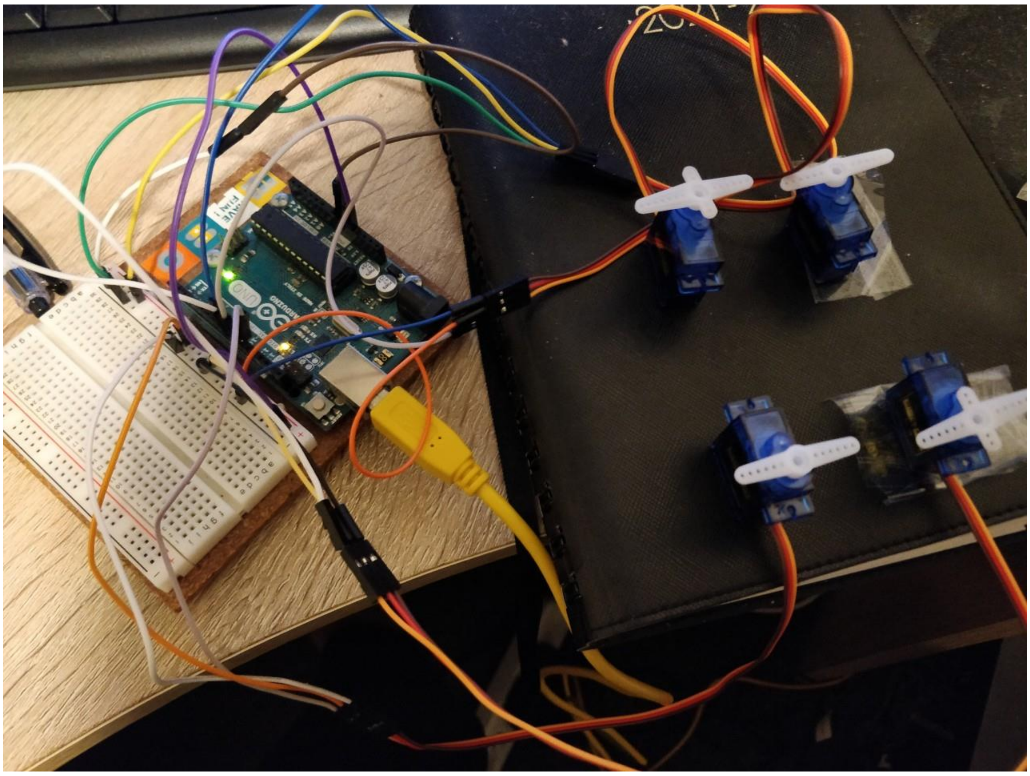


Figure 3.24 – Servo Control using Arduino UNO

3.4.2 Testing circuit

Once the components were selected and schematic diagram was drawn, the circuit test was conducted before assembling all the components to the perfboard. I used the bread board to set up the buck converter and the voltage regulator to confirm the output voltages. Also, this test was to see if the circuit was correctly designed for other main components to work correctly. For example, ESP32 requires stable 3.3V DC input and without the correct decoupling capacitor, it will have the brown out problem which causes the Bluetooth and Wi-Fi features on ESP32 unavailable. The circuit was designed so the supply voltage will be stepped down by the Buck converter to 5V which will supply the servo motors and L298 IC. The stepped down 5V will then be regulated to 3.3V by the LD1117 for the ESP32. The figure 3.25 shows the testing of the voltage modules. I also did an experiment of placing the wrong capacitor which caused the LED to flicker and the noticeable noise was heard. By having the correctly chosen components, the LED stayed on steady and no noise was heard.

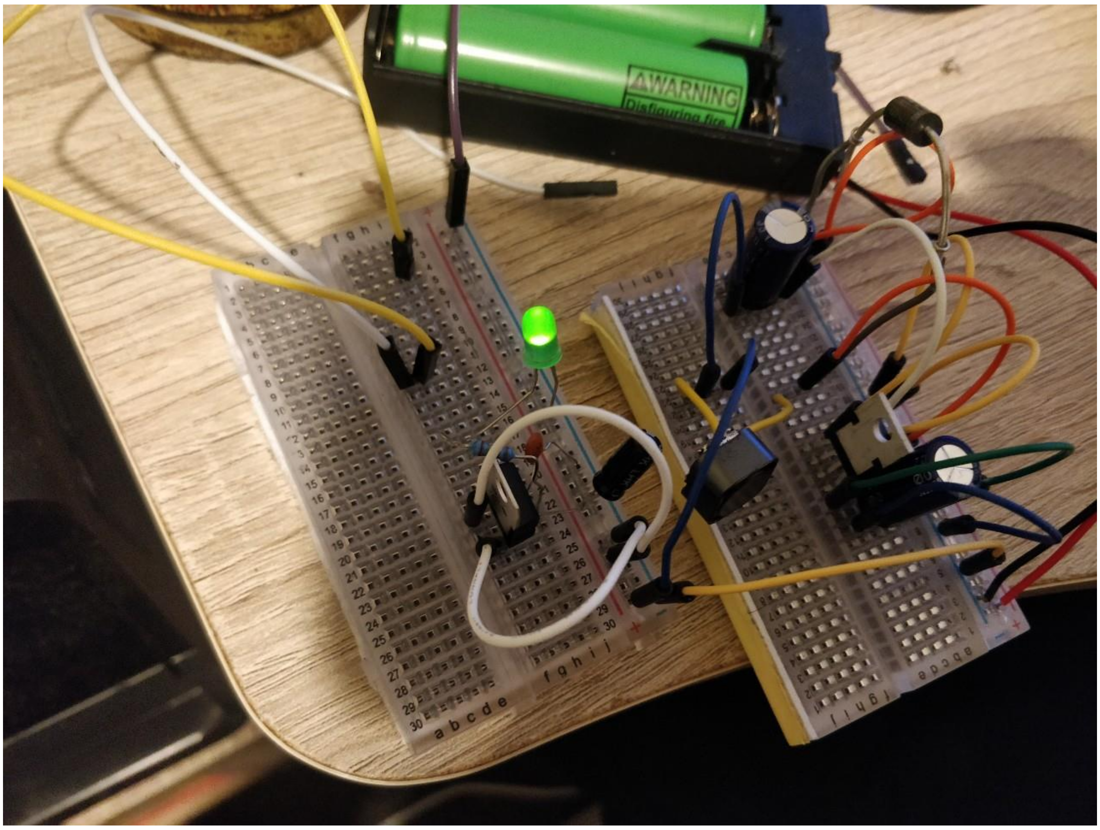


Figure 3.25 – Testing Buck converter and Voltage regulator

3.4.3 Testing PCB

After the testing circuit with bread board, I soldered all the components to the perfboard. I carried out the continuity check using a multi meter each time I soldered to make sure all the connections are done right and making sure the ground and supply wires are connected properly. Once all the components were soldered on, input was supplied by the batteries and measured the output pins of the buck converter and voltage regulator. There are LEDs placed for each power supplies to indicate that those are working properly.

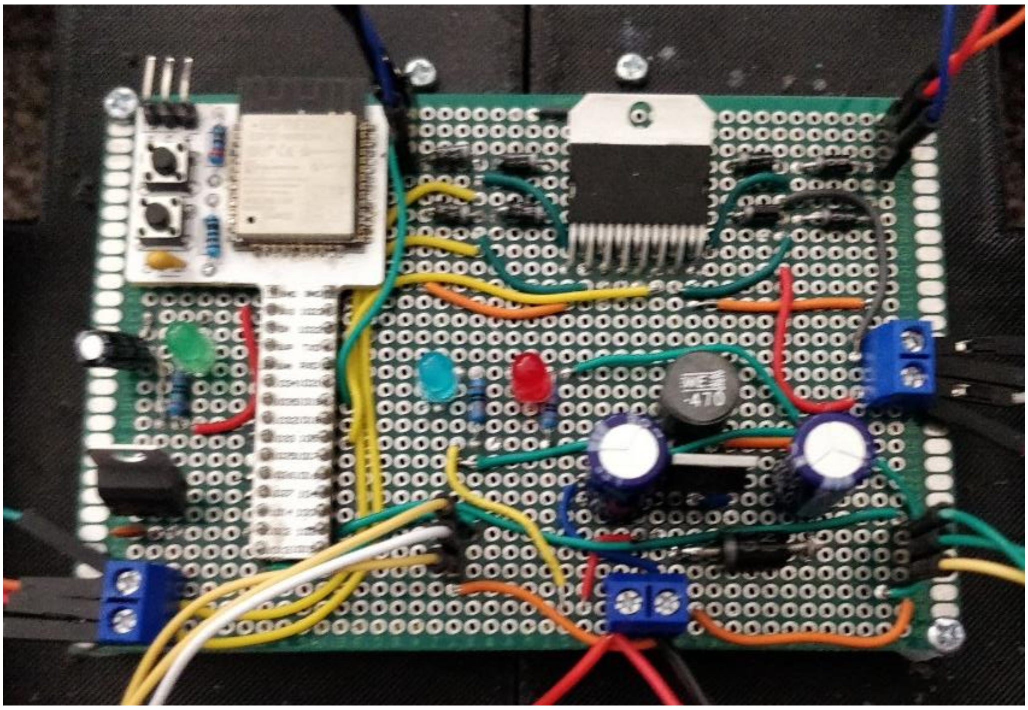


Figure 3.26-Assembled PCB

3.5 Evaluation

On completion of designing app, circuits, body and assembling the PCB, it was time to evaluate the performance of the prototype. By supplying the power, all three LED lights were on and the Bluetooth connection was successful with the app. The prototype was able to move and perform all the modes of steering as the user commands however, due to the basic DC motor with its low power, there were no noticeable of the robot gradually speeding up. The top speed when coding was 255 and the motor was not strong enough to move when the peed was 180. The difference in these speeds were very unnoticeable. This feature of speeding up can be used once there is an upgrade to powerful motor.

During the evaluation phase, I also asked my family to try out using the robot. There was common feedback of the controlling was difficult as the app was designed in portrait orientation and a person with bigger hands have more disadvantage. Otherwise they all agreed that it was easy to use and understand how to give commands for various modes of steering.

3.5.1 Battery Indicator

This is a battery operated project and there were no indications for the users to know about the battery level. Knowing that the minimum voltage required by the LM2596 is 4.5V, all the motors will stop working once the battery goes below 5V. During evaluation phase, I was able to pick up this feedback and implemented the simple LED and resistor circuit to the prototype.

The standard red, yellow and green LEDs were used which has voltage drop of around 2V. In theory, by connecting these in series all three should be lit up when more than 6V is supplied and two LEDs should be on below 6V. This way the user can be informed by the number of LEDs that were on. Therefore, when there are only two LEDs lit, it is indicating that the battery is supplying less than 6V and needs charging.

Figure 3.27 shows the circuit diagram of the LED battery indicator.

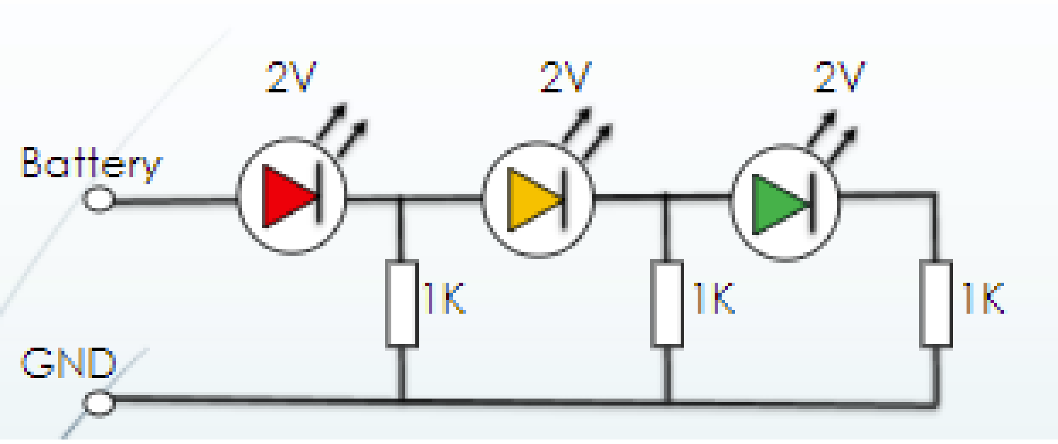


Figure 1.27 – Circuit diagram for battery indicator

3.5.2 Ackermann Steering Geometry

When a vehicle is turning around a circle, the angles of inner and outer wheels need to be different to avoid tipping or slipping of the vehicle. Ackermann steering geometry allows us to find these angles around a certain center point. The angles can be found using a formula or manually drawing.

The angles can be found using the formula and parameters shown below:

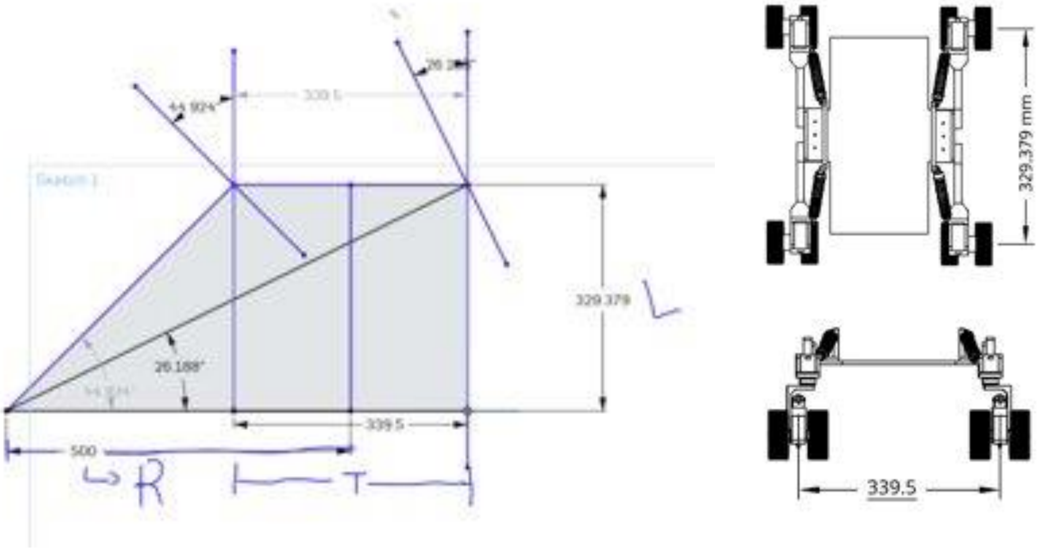


Figure 3.28 – parameters for finding angles

Tan a(inner) = L/(R-(T/2))

Tan a(outer) = L/(R-(T/2))

Chapter Four: Results and Findings

4.1 Prototype Result

The overall construction of prototype was completed and was able to control with a smartphone using the app that I designed. The objective of providing various steering modes were met and this prototype can move freely in many directions. The prototype PCB is slightly bigger than Arduino Uno board, but it was the half size of an average smartphone. It is also designed so that it can have a supply voltage up to 24V under $50 which means that the prototype can be easily upgraded to much powerful prototype without changing any parts of the PCB.

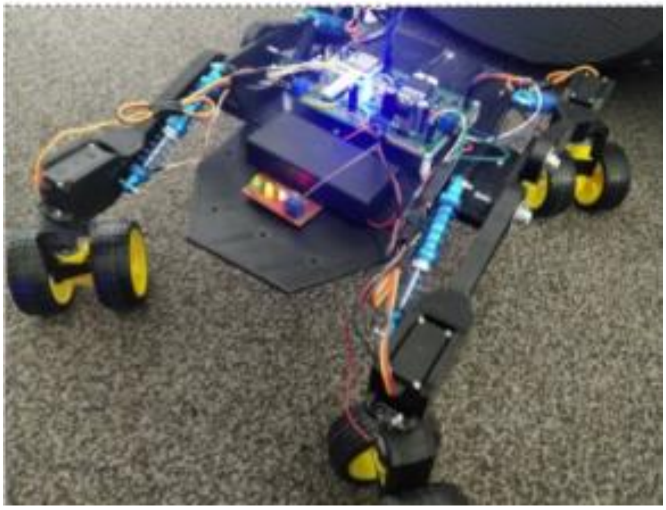


Figure 2- Completed Prototype

4.2 Battery Indicator Result

Once circuit diagram was designed for the indicator, I have used a vero board to assemble components and tested the board with a multi meter. Due to not having 1K resistors, I used two 510 ohm resistors in series for each LED.

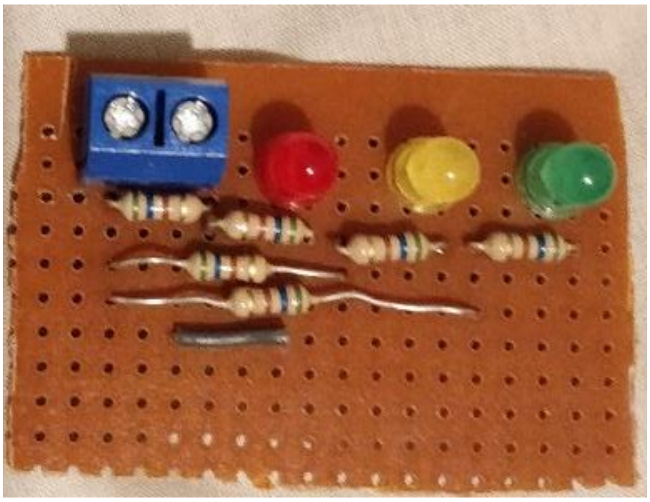


Figure 4.1-Soldered Battery indicator

The result for the behaviour of LEDs were shown as expected. All three lights are on when fully charged, and two LEDs are fully on until around 5.5V and finally one LED is on below 4V.

Figures 4.2, 4.3 and 4.4 below shows the testing of the indicator by using potentiometer to alter input voltage.

Figure 4.2 – Fully charged

Figure 4.3 – Needs Charging

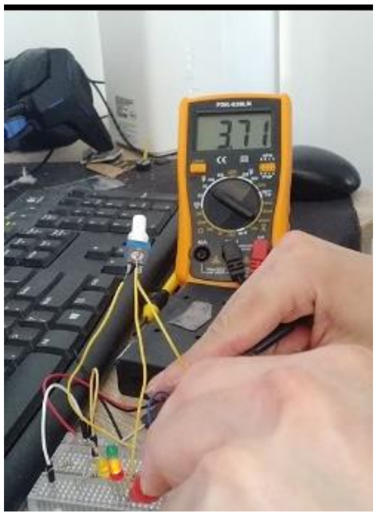


Figure 4.4 – No power for robot

4.3 Steering Results

The concept of various steering option was successfully achieved by using servo motors for each wheel. Figures below are the photos of prototype showing different wheel position for each mode.

Figure 4.5 – Straight Mode

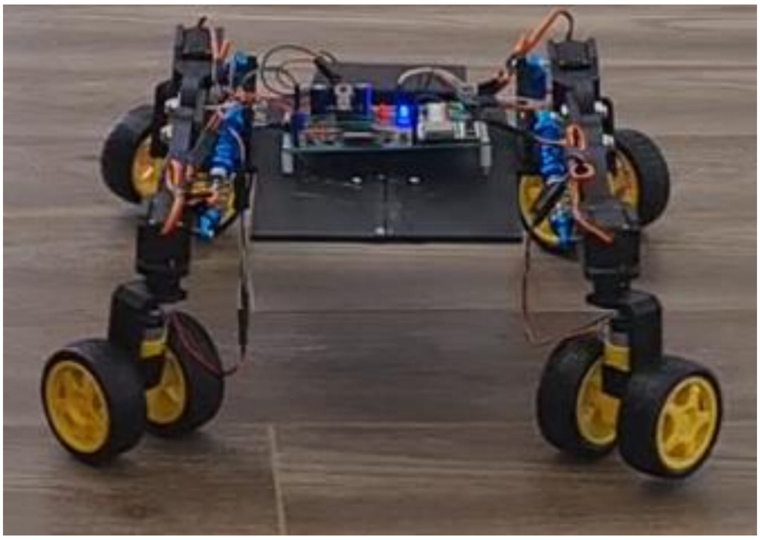


Figure 4.6 – 360 Rotation Mode

Figure 4.7 – 180 Horizontal Mode

Figure 4.8 – Crab Mode

The angles found for the turning was implemented to my prototype and the distance from center of the vehicle to the center point was set at 500mm. This distance can be changed as required. The robot was able to turn around the center point shown by the electrical tape. The angles calculated were correct as the robot was going around the circle without going off the radius of 500mm. The angle for the inner wheel was 44.9 degrees and outer wheel was 63.9 degree. These values were put in to the line of code for servos. The figure 4.9 shows the testing of the Ackermann's steering geometry.

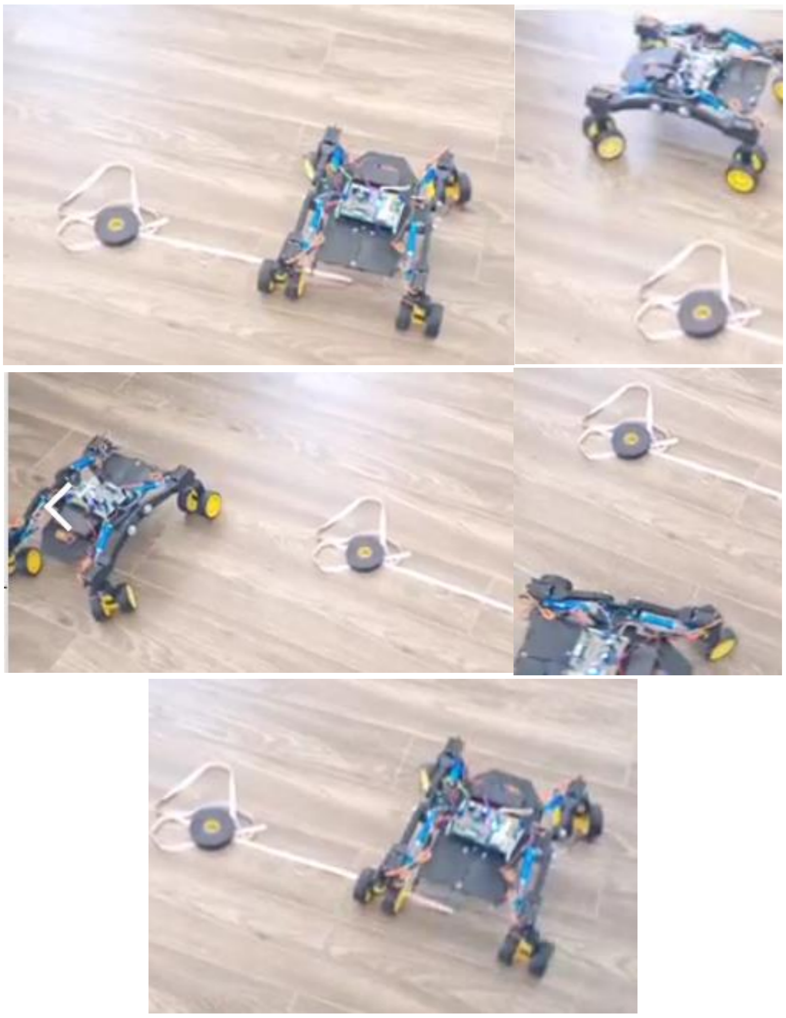


Figure 4.9- Testing of Ackermann Steering Geometry

Chapter Six: Conclusion

6.1 Conclusion

The aim of this project was to design and construct a prototype robot to support another project which was LED plant detecting system. These projects put together was to provide weed detecting agricultural robot to help in the agriculture. The prototype I have created aim to be used with many other projects and in many industries other than agriculture.

This prototype robot is controlled remotely and can move in many direction using four servo motors for each wheel. This project shows that a low cost PCB that is half the size of a smartphone can deliver the performance that is required of a vehicle platform. The PCB only costed under 50 dollars to build and it can have supply voltage of up to 24V and output of maximum of 3A.

There are many advantages using robots in many industries including the agriculture. It is developed to solve issues of lacking workers, physical labouring and safety of people. My prototype was designed for easy use and leaves room for easy modification of hardware and software for adding more complex features.

6.2 Future Work

There are many development for this prototype. Main future work will be to upgrade the DC motors, batteries and chassis of the prototype to aluminium for bigger platform. ESP32 has Wi-Fi features which can be developed into controlling remotely by a computer or a laptop instead of a smart phone. Adding a camera to the protype will allow the users to control inside at a workstation while the robot is outside. Wi-Fi gives a much greater range than the Bluetooth which is very advantageous. When this prototype is upgraded with more powerful motors and bigger chassis, it can be tested with the other projects like the LED weed detecting system.

References

Association, R. I. (2017, December 26). *Robots' benefits in the agricultural industry*. Retrieved from ControlEngineering: https://www.controleng.com

Bak, L. (2020, December 9). *How robots make agriculture more sustainable*. Retrieved from Innovationorigins: https://innovationorigins.com

Editorial. (2019, July 18). *Robotics in agriculture advantages nad disadvantages*. Retrieved from Roboticsbiz: https://roboticsbiz.com

Harris, C. (2020, January 21). *Stuff*. Retrieved from Robot start-up Radius Robotics seeks to solve world's soil depletion: http://www.stuff.co.nz

Molloy, N. (2021, April 29). *Agricultural-robotics*. Retrieved from New Zealand's Vision for Agricultural Robotics: http://www.agricultural-robotics.com

N.Vamshidhar Reddy, A. V. (2016, July 4). *A critical review on agricultural robots.* Retrieved from ResearchGate.

Robot, M.-1. A. (2012, May 1). *shvachko*. Retrieved from http://www.shavchko.net

Wilsong, L. (2019, August 30). *Quad bike injury data raises questions about children, older riders*. Retrieved from Stuff: https://www.stuff.co.nz

Appendices

Appendix 1: Graduate Attributes, Definitions and Evidences of Achievement

Ordering & quality | Technical documentation | Design & development | Support & training

TEXAS INSTRUMENTS

LM2596

SNVS124F – NOVEMBER 1999 – REVISED APRIL 2021

LM2596 SIMPLE SWITCHER® Power Converter 150-kHz 3-A Step-Down Voltage Regulator

1 Features

- New product available: LMR33630 36-V, 3-A, 400-kHz synchronous converter
- 3.3-V, 5-V, 12-V, and adjustable output versions
- Adjustable version output voltage range: 1.2-V to 37-V ±4% maximum over line and load conditions
- Available in TO-220 and TO-263 packages
- 3-A output load current
- Input voltage range up to 40 V
- Requires only four external components
- Excellent line and load regulation specifications
- 150-kHz Fixed-frequency internal oscillator
- TTL shutdown capability
- Low power standby mode, I_Q, typically 80 μA
- High efficiency
- Uses readily available standard inductors
- Thermal shutdown and current-limit protection
- Create a custom design using the LM2596 with the WEBENCH Power Designer

2 Applications

- Appliances
- Grid infrastructure
- EPOS
- Home theater

3 Description

The LM2596 series of regulators are monolithic integrated circuits that provide all the active functions for a step-down (buck) switching regulator, capable of driving a 3-A load with excellent line and load regulation. These devices are available in fixed output voltages of 3.3 V, 5 V, 12 V, and an adjustable output version.

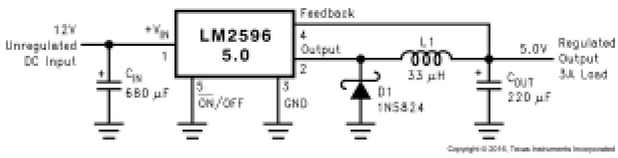
Requiring a minimum number of external components, these regulators are simple to use and include internal frequency compensation, and a fixed-frequency oscillator.

The LM2596 series operates at a switching frequency of 150 kHz, thus allowing smaller sized filter components than what would be required with lower frequency switching regulators. Available in a standard 5-pin TO-220 package with several different lead bend options, and a 5-pin TO-263 surface mount package.

The new product, LMR33630, offers reduced BOM cost, higher efficiency, and an 85% reduction in solution size among many other features. Start WEBENCH Design with the LMR33630.

PART NUMBER	PACKAGE[1]	BODY SIZE (NOM)
LM2596	TO-220 (5)	14.986 mm × 10.16 mm
	TO-263 (5)	10.10 mm × 8.89 mm

(1) For all available packages, see the orderable addendum at the end of the data sheet.



(Fixed Output Voltage Versions)

Typical Application

An IMPORTANT NOTICE at the end of this data sheet addresses availability, warranty, changes, use in safety-critical applications, intellectual property matters and other important disclaimers. PRODUCTION DATA.

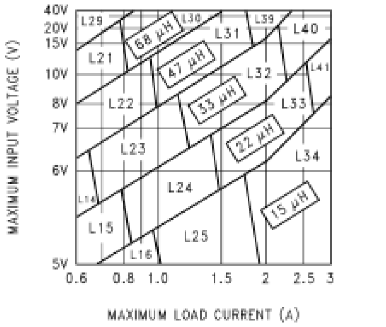


Figure 9-5. LM2596-3.3

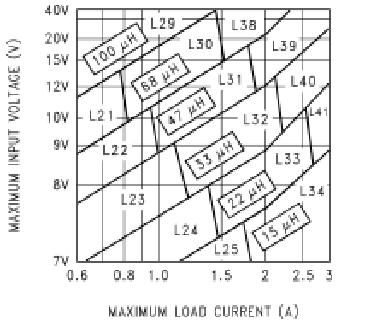


Figure 9-6. LM2596-5.0

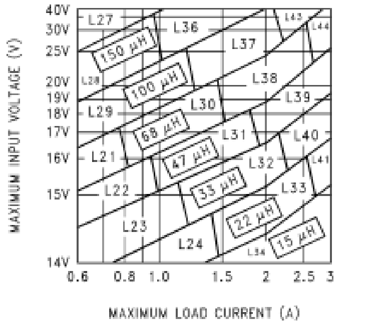


Figure 9-7. LM2596-12

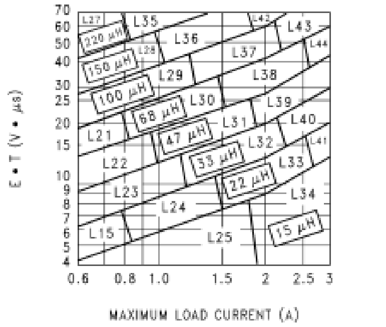


Figure 9-8. LM2596-ADJ

Table 9-1. Inductor Manufacturers Part Numbers

	INDUCTANCE (µH)	CURRENT (A)	SCHOTT		RENCO		PULSE ENGINEERING		COILCRAFT
			THROUGH-HOLE	SURFACE-MOUNT	THROUGH-HOLE	SURFACE-MOUNT	THROUGH-HOLE	SURFACE-MOUNT	SURFACE-MOUNT
L15	22	0.99	67148350	67148460	RL-1284-22-43	RL1500-22	PE-53815	PE-53815-S	DO3308-223
L21	68	0.99	67144070	67144450	RL-5471-5	RL1500-68	PE-53821	PE-53821-S	DO3316-683
L22	47	1.17	67144080	67144460	RL-5471-6	—	PE-53822	PE-53822-S	DO3316-473
L23	33	1.40	67144090	67144470	RL-5471-7	—	PE-53823	PE-53823-S	DO3316-333
L24	22	1.70	67148370	67148480	RL-1283-22-43	—	PE-53824	PE-53825-S	DO3316-223
L25	15	2.10	67148380	67148490	RL-1283-15-43	—	PE-53825	PE-53824-S	DO3316-153
L26	330	0.80	67144100	67144480	RL-5471-1	—	PE-53826	PE-53826-S	DO5022P-334
L27	220	1.00	67144110	67144490	RL-5471-2	—	PE-53827	PE-53827-S	DO5022P-224
L28	150	1.20	67144120	67144500	RL-5471-3	—	PE-53828	PE-53828-S	DO5022P-154
L29	100	1.47	67144130	67144510	RL-5471-4	—	PE-53829	PE-53829-S	DO5022P-104
L30	68	1.78	67144140	67144520	RL-5471-5	—	PE-53830	PE-53830-S	DO5022P-683
L31	47	2.20	67144150	67144530	RL-5471-6	—	PE-53831	PE-53831-S	DO5022P-473
L32	33	2.50	67144160	67144540	RL-5471-7	—	PE-53932	PE-53932-S	DO5022P-333

Table 9-3. LM2596 Fixed Voltage Quick Design Component Selection Table

CONDITIONS			INDUCTOR		OUTPUT CAPACITOR			
					THROUGH-HOLE ELECTROLYTIC		SURFACE-MOUNT TANTALUM	
OUTPUT VOLTAGE (V)	LOAD CURRENT (A)	MAX INPUT VOLTAGE (V)	INDUCTANCE (µH)	INDUCTOR (#)	PANASONIC HFQ SERIES (µF/V)	NICHICON PL SERIES (µF/V)	AVX TPS SERIES (µF/V)	SPRAGUE 595D SERIES (µF/V)
3.3	3	5	22	L41	470/25	560/16	330/6.3	390/6.3
		7	22	L41	560/35	560/35	330/6.3	390/6.3
		10	22	L41	680/35	680/35	330/6.3	390/6.3
		40	33	L40	560/35	470/35	330/6.3	390/6.3
		6	22	L33	470/25	470/35	330/6.3	390/6.3
	2	10	33	L32	330/35	330/35	330/6.3	390/6.3
		40	47	L39	330/35	270/50	220/10	330/10
5	3	8	22	L41	470/25	560/16	220/10	330/10
		10	22	L41	560/25	560/25	220/10	330/10
		15	33	L40	330/35	330/35	220/10	330/10
		40	47	L39	330/35	270/35	220/10	330/10
		9	22	L33	470/25	560/16	220/10	330/10
	2	20	68	L38	180/35	180/35	100/10	270/10
		40	68	L38	180/35	180/35	100/10	270/10
12	3	15	22	L41	470/25	470/25	100/16	180/16
		18	33	L40	330/25	330/25	100/16	180/16
		30	68	L44	180/25	180/25	100/16	120/20
		40	68	L44	180/35	180/35	100/16	120/20
		15	33	L32	330/25	330/25	100/16	180/16
	2	20	68	L38	180/25	180/25	100/16	120/20
		40	150	L42	82/25	82/25	68/20	68/25

The capacitor list contains both through-hole electrolytic and surface-mount tantalum capacitors from four different capacitor manufacturers. TI recommends that both the manufacturers and the manufacturer's series that are listed in Table 9-3.

In this example aluminum electrolytic capacitors from several different manufacturers are available with the range of ESR numbers required.

- 330-µF, 35-V Panasonic HFQ Series
- 330-µF, 35-V Nichicon PL Series

3. The capacitor voltage rating for electrolytic capacitors should be at least 1.5 times greater than the output voltage, and often require much higher voltage ratings to satisfy the low ESR requirements for low output ripple voltage.

   For a 5-V output, a capacitor voltage rating of at least 7.5 V is required. But even a low ESR, switching grade, 220-µF, 10-V aluminum electrolytic capacitor would exhibit approximately 225 mΩ of ESR (see Figure 9-2 for the ESR versus voltage rating). This amount of ESR would result in relatively high output ripple voltage. To reduce the ripple to 1% or less of the output voltage, a capacitor with a higher value or with a higher voltage rating (lower ESR) must be selected. A 16-V or 25-V capacitor will reduce the ripple voltage by approximately half.