Computer Aided Design and Prototyping

(ME444 – FALL 2022)

[Starship Transformer Robot]

Design Report

Group No. 19

Reuben Irizarry	Senior - ME
Jordan Busard	Senior - ME
Daniel Bergman	Senior - ME
Abhishek Nair	Senior - ME

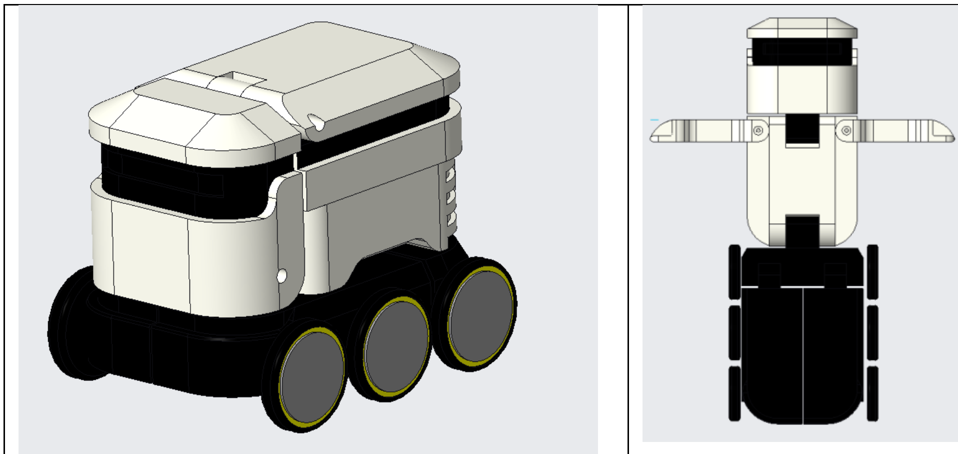
School of Mechanical Engineering
Purdue University

Revision History

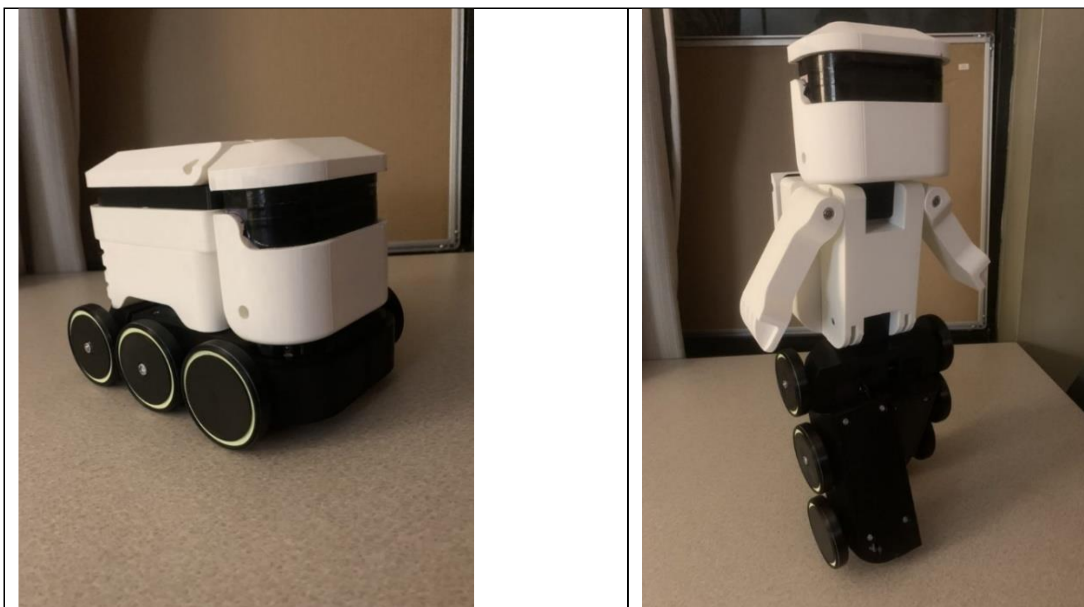
S. No.	Date	Revision ID	Revision Details (Page No., Paragraph, Line No. etc.)	Author
1.	11/30/22	Revision A	Initial Release	Jordan Busard
2.	12/5/22	Revision B	Introduction	Reuben Irizarry
3.	12/9/22	Revision C	Finalization of Report	Daniel Bergman, Jordan Busard, Abhishek Nair

Executive Summary

The Starship Transformer Robot is a controlled miniature starship robot that may also transform into humanoid form. The concept for this robot was inspired by the iconic Starship robots seen throughout the Purdue University campus as well as the 2007 hit movie "Transformers". The robot is equipped with 2-wheel driving capability and differential steering alongside headlights and Starship voice-lines. This Starship can also be fully controlled from a phone!



Pictures of Toy Assembly Rendered in Creo



Picture of the Physical Prototype of the Toy

Note: specified here is the file name and location of the final toy assembly in executive summary (me444g19/CAD/transformer).

Table of Contents

1. Introduction
    1.1. Project Backgrounds
    1.2. Design Requirements and Constraints

2. Concept Generation and Selection

3. Detail Design
    3.1. Transformation Joints
    3.2. Drive Train
    3.3. Electronic Systems
    3.4. System Integration

4. Prototyping and Testing

5. Results and Discussion

6. Conclusion

7. References

8. Appendices

1. Introduction

1.1 Project Backgrounds

The concept for this robot was inspired by the iconic Starship robots seen throughout the Purdue University campus as well as the 2007 hit movie “Transformers”.

The starship robot is equipped with 2-wheel drive and differential steering. The robot can be remote controlled via the Blynk app on the user’s phone. In addition to the core components, the robot has headlights and the signature Starship sound.

The target age group for the toy is age 7+. Due to the electrical components and size of the toy, a younger child will likely not enjoy the full value of the robot. In addition, younger children could be at risk of injury due to small components.

As time goes on, children are becoming more and more familiar with technology and robots. This product is very accessible, combining the appeal of a robot with that of an action figure. Our product has a very large market potential, especially for children in and around college campuses that have Starship robots. Many children are fascinated with these robots and would likely be very eager to have a smaller version at home.

Currently, the closest competitor to this product are action figures related to the Transformer movies that collapse into cars. However, this specific robot would appeal to those who are familiar with Starships and their notoriety.

1.2 Design Requirements & Constraints

Design Requirements:

- New toy with marketing potential
- Significantly different features than current toys available in the market
- Mechanical Toy with enough complexity
    - Mechanical functionalities with at least two coordinated motions
- Maximum size of 10” x 10” x 10”

- $60 budget

All of the imposed constraints guided our design. Due to our product changing in size, the size requirement is met when the robot is fully collapsed. We were well under the budget constraint, only having to spend 15 dollars on an MP3 speaker module, as we already had all the necessary 3D printing filament and fasteners. Our original idea was for the Starship Transformer robot, so the mechanical functionality requirements were inherently met.

2. Concept Generation and Selection

When generating concepts for this project, Starships was a certainty for our design, However, in order to put a unique twist on an RC-controlled delivery robot, we came up with the idea of making it transform into a humanoid form. Much of our planning for this design was around how the toy would collapse, and a few different iterations of the folding mechanism were tossed around.

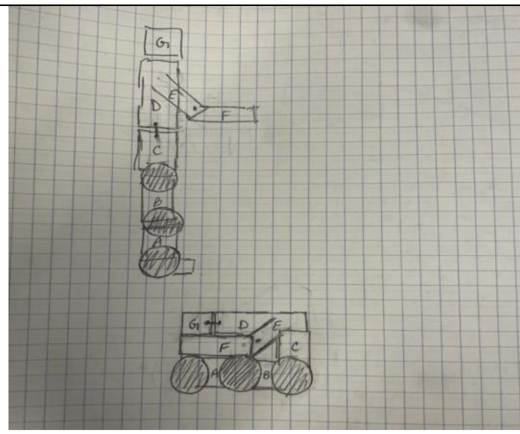


Figure 2.1: Initial Concept Drawing

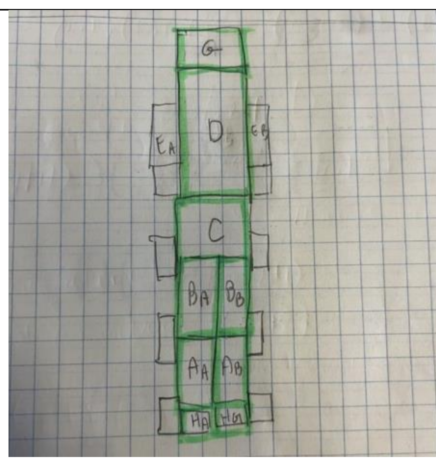


Figure 2.2: Later Concept Drawing

After some iteration on the CAD, the design of the different folding segments of the toy became even further simplified.

3. Detail Design

3.1 Transformation Joints

The structure of the toy, which consisted of seven major parts connected by a total of 6 pin joints, allowed the toy to fold between its two forms. The seven parts were a head piece, a body piece, two arms, a hip, and two legs.

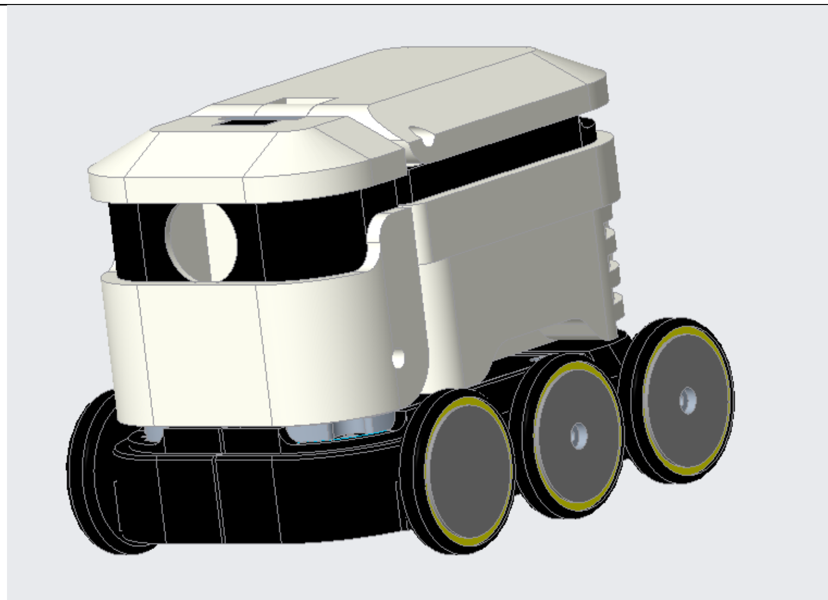


Figure 3.1.1: Full Body (file: *me444g19/CAD/transformer)*

The body, the main part of the assembly and also the largest part, featured a large cavity, akin to Starship's large cavity used for storage of goods. This cavity was used for storage of electronics in our design. On the body there were 4 joints. One connecting to each arm, one which was connected to the head using an additional neck piece, and then a joint connecting to the hip.

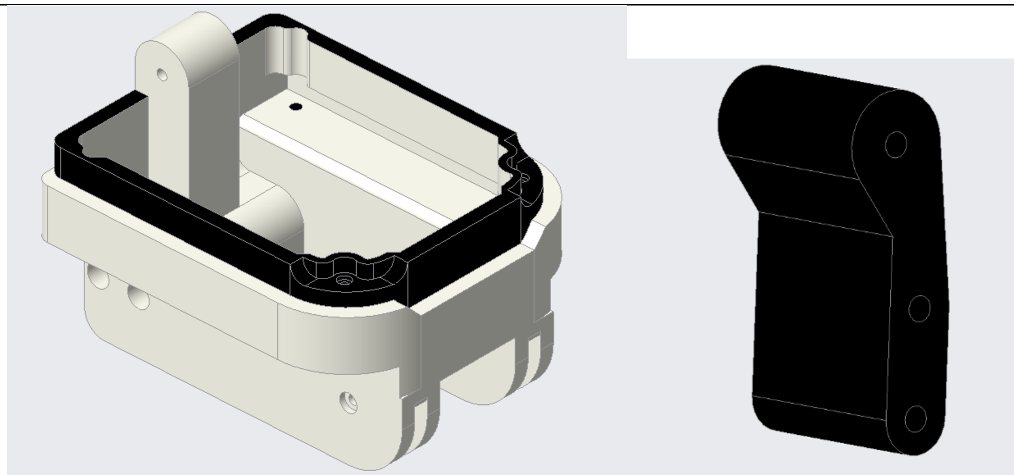


Figure 3.1.2: CAD Models of Body and Neck Pieces (file: *me444g19/CAD/neck and me444g19/CAD/d)*

The head, which connected to the neck piece from the body, also housed the sound system, which will be explained further in section 3.3. A hole was created in the front of the head to allow for easier amplification of the produced sounds. Incidentally, this hole somewhat appears like an eye, adding to the transformer aesthetic. The head was designed to have exactly 90 degrees of freedom.

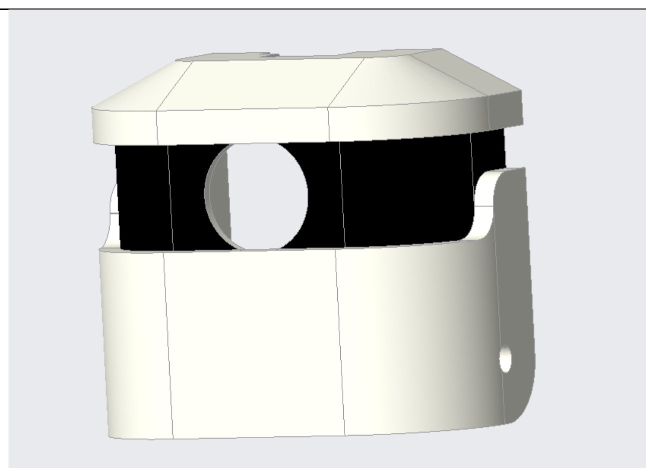


Figure 3.1.3: CAD Model of Head Piece (file: *me444g19/CAD/g)*

The arms, which were mirrored parts, mounted from the body and also had around 90 degrees of freedom. The arms were modelled to look as if the robot had hands, to help better visualize the robot in humanoid form.

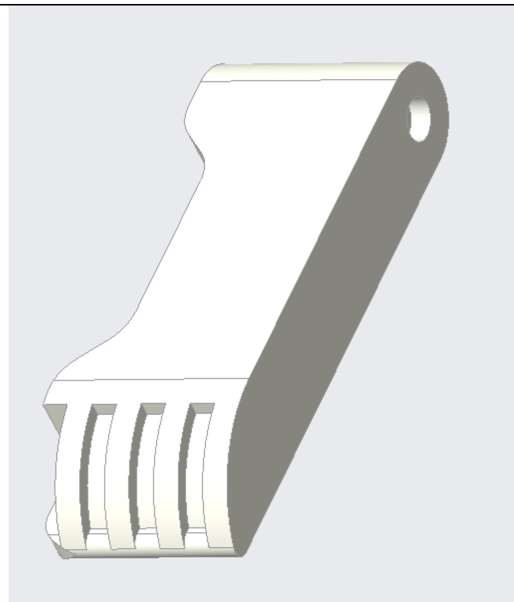


Figure 3.1.4: CAD Model of Arm Piece (file: *me444g19/CAD/e_b)*

The hip piece was important to add an additional 90 degrees of rotation to the robot, allowing it to stand completely. The body attached to the hip with a single pin joint. The hip also attached to each leg with a pin joint, and featured holes for mounting idler wheels for the drive train, as will be mentioned in section 3.2.

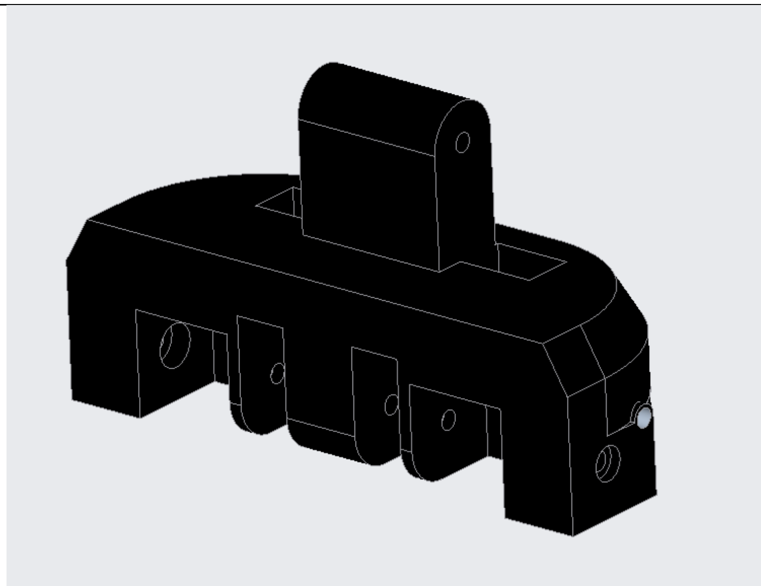


Figure 3.1.5: CAD Model of Hip Piece (file: *me444g19/CAD/c_1)*

3.2 Drive Train

The drive train of the toy was a relatively simple in concept mechanism. To match a starship, the toy was designed with six wheels. Housed inside the legs are motors which drive the front two wheels on the toy. The other four wheels free spin on a screw and are slick to allow the robot to turn easier. The electronics behind the drive train are explained more in section 3.3.

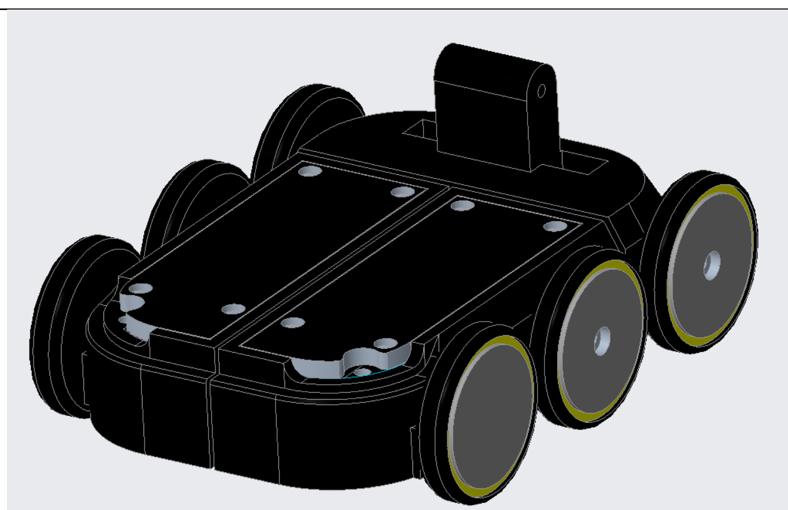


Figure 3.2.1: CAD Model of Only Drive Train

3.3 Electronic Systems

The electronic system consisted of a 9V battery, switch, ESP32 microcontroller and a motor controller board. This microcontroller was controlled using Blynk, a cloud-based control software which used Wi-Fi connection to send and receive signals between a cellular phone and the microcontroller. Due to a hardware issue in which the microcontroller did not have a functional voltage input port, the toy requires being plugged into a USB power source in order to drive and for the lights to function. Shown below is a circuit diagram of the electronics system. The system also featured 4 LEDs to emulate the front and rear lights of a Starship robot, each wired to a 5V source.

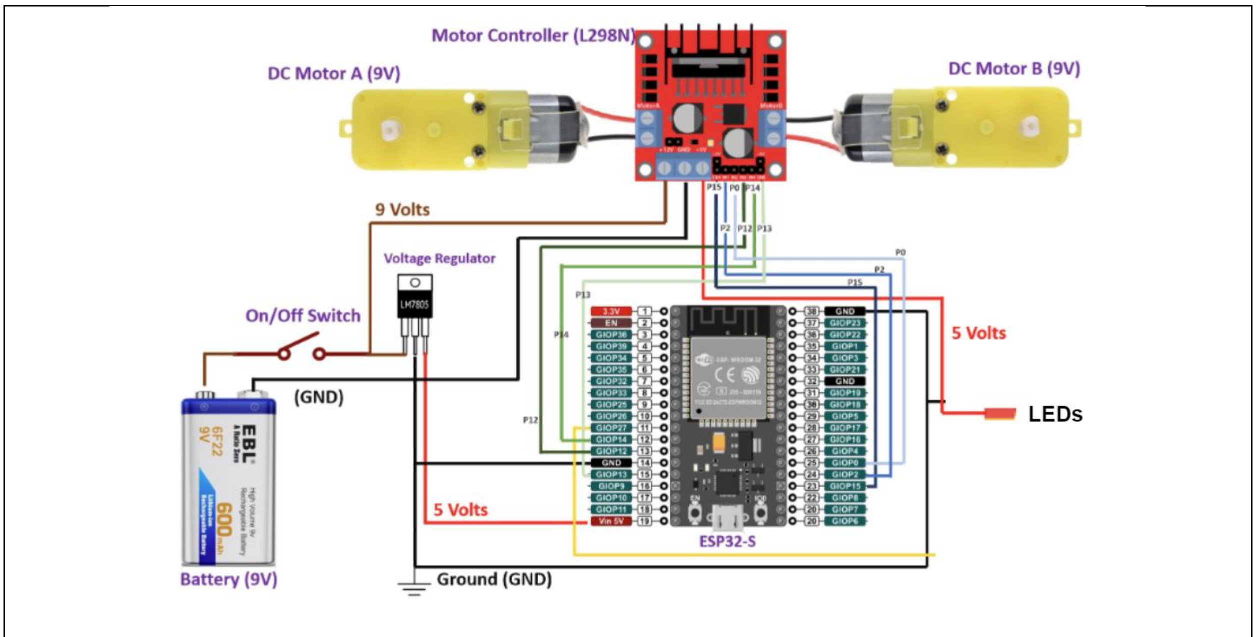


Figure 3.3.1: Circuit Diagram

In addition to this electronic system, another system was housed inside the head in order for the toy to make noise. A battery pack, speaker, button, and small circuit, which were purchased as a single unit was used as the sound system for the toy. Shown below is an image of this sound system.

Figure 3.3.2: Sound System

3.4 System Integration

In order to design the prototype, a single CAD model of a Starship delivery robot was first made in the desired scale of the toy. Then, the individual folding segments were cut from the design, such that the single model became an assembly of blocks that together formed a Starship robot. Then, joints were added to allow the toy to change form appropriately. Next, locations for electronics were finalized. Finally, aesthetic and minor details were ironed out prior to manufacturing the toy.

4 Prototyping and Testing

Prototyping was conducted through 3D printing, with issues being noted while assembling the design and rectified within CREO. Fit and tolerances were tested by assembling parts of the prototype and testing whether they mated as required and/or moved as required (for joints).

We applied DFM concepts in order to reduce the complexity of the final design while maintaining the functionalities we decided within concept generation. In 3D printing most of our components and designing the wire routing and pin joints as integrated parts of the CAD model, we saved a significant amount of money as well as further prototypes and tooling costs. Parts could not be combined with one another in order to simplify the design, since the toy being a transformer required manual actuation of several parts in order to form the limbs and the head of the toy, thus also requiring joints for the same. The pin joints were also constructed using off-the-shelf bolts, simplifying the design of and reducing costs.

We were not able to apply DFA concepts in designing our robot due to space constraints in fitting much of the electronics within the cavities within each body section of the robot, as well as to rout wires between each part for connectivity and power. Some parts, such as the LEDs were also extremely small and difficult to maneuver around, but this problem could be mitigated by using parts that are better suited for the constrained spaces within the robot. Wiring could also be improved by using a different solution than a breadboard and standardized jumper cables. This would increase the usable space within the cavities and potentially allow for storage of small items similar to a real Starship robot.

As previously mentioned, we already had most of the items required in creating the robot, except for a speaker module, which was purchased for $15. The full bill of materials is as follows:

BOM

	Vendor	Item
1	Amazon	Yeeco DC Gear Motor for Smart Car Robot
2	Amazon	Qunqi L298N Motor Drive Controller Board Module Dual H-Bridge DC Stepper for Arduino

3	**Amazon**	**AITRIP ESP32 ESP-32 Development Board 2.4 GHz WiFi + Bluetooth 2-in-1 Microcontroller**
4	**Amazon**	**VONVOFF Toggle Switch,ON/Off SPST with Pre-soldered**
5	**Amazon**	**EBL Rechargeable 9V Batteries Li-ion 9V Battery**
6	**Amazon**	**ZYAMY 400 Tie Point Prototype Solderless PCB Breadboard**
7	**Amazon**	**9 V Battery Clip Connector, T-Type 9 Volt Battery Clip**
8	**Amazon**	**L7805 Positive Voltage Regulator Output 5v**
9	**RoboSource**	**#8-32 Screws and Nuts**
10	**Amazon**	**Talking Products MP3 sound module**

5. Results and Discussion

After the prototype was fully assembled and electronics were set up as desired, all functionality of the toy was tested.

The sound function worked very well. During our demonstration at the toy fair, it was difficult to hear, but it was loud enough to catch everyone's attention in a quiet classroom or if the robot made a noise in another quiet public space. An additional functionality that could be implemented to the sound system could be a volume knob, such that it could be turned up for demonstration in crowded areas or turned down very low for situations where the sound may be a distraction to others.

Most of the joints for the transformation between Starship and Transformer performed as intended. The joints associated with the hip piece were the only troublesome joints. Where the legs connected to the hips, there was too much interference, resulting in an unpleasant noise when folding the legs. This interference however was helpful for putting the robot in a standing stance, as if they were loose the robot would likely fall over. For this, using a stiffer joint with little to no interference would likely be much better and would perform similarly.

Sadly, due to electronics issues we were never able to see the lights and drive in full function. The ESP32 microcontroller that we had did not have a functioning 5V power supply, limiting its power source to only be its micro-USB port. When powered through micro-USB, the drive was fully functional, but sluggish. The drive also had performance issues on slick surfaces. Given more prototyping time, more care could have been made to get the electronics system running smoothly. The lights could also have been powered by a source separate from the microcontroller and the drive so that they could be functional even if the robot is not being driven by remote control.

Two large learning outcomes can be taken away from the results of the prototype. First, rarely will the first prototype be perfect. Sometimes the twentieth prototype may not be perfect. Finding flaws in a design of a product is a necessary and inevitable part of prototyping. Second, although the mechanical design of the toy made up a large amount of its functionality, the electronics implementation should not be ignored. Especially when many separate functions are running from one system, things can easily go wrong. With adequate time allocated to electronics debugged, we likely could have had full electronic function of our toy.

6. Conclusion

In summation, we were able to design a transforming Starship robot able to fold out into a design greatly resembling the humanoid forms of the popular Transformers toys available in stores, while maintaining the ability to wirelessly locomote in its folded Starship form. We were also able to implement lighting and sound. Challenges were noted in its design process due to component failure within the ESP32 microcontroller and lack of power for the system. In the future, the robot may be improved by adding a robust power source as well as a different ESP32 board to ensure proper electronic functionality. We could also possibly add further functionality in the sound and light systems to emulate Starship robots to a closer degree, or potentially evolve the design for the parts to self-actuate and transform without external intervention.

7. References

N/A

8. Appendices

8.1 Arduino Code

```
#define BLYNK_PRINT Serial


#define BLYNK_TEMPLATE_ID "TMPL5AHqPJP0"
#define BLYNK_DEVICE_NAME "RC CAR ME 444"
#define BLYNK_AUTH_TOKEN "atQMvrva03TuRi-2aRqcxOPzwTChzoCR"


#include <BlynkSimpleEsp32.h>


char auth[] = "atQMvrva03TuRi-2aRqcxOPzwTChzoCR";


#include "ESP32Servo.h"


int servoPin = 27;     //specify pin numbers according to the circuit diagram
int servoAngle = 90;
int enablePinA = 15;
int in1PinA = 2;
int in2PinA = 0;
int enablePinB = 13;
int in1PinB = 12;
int in2PinB = 14;


Servo servo1;  //claim the servo motor object


void setup()
{
  Serial.begin(115200);
  delay(100);



  Blynk.begin(auth, "NETGEAR39", "phobicwater900");
```

```
  servo1.attach(servoPin);    //set up the servo pin as pin 27
  servo1.write(servoAngle);    //initialize the servo motor at 90 degree.
  pinMode(in1PinA, OUTPUT);    //set in1PinA(2) as an output pin
  pinMode(in2PinA, OUTPUT);    //set in2PinA(0) as an output pin
  pinMode(enablePinA, OUTPUT); //set enablePinA(15) as an output pin
  pinMode(in1PinB, OUTPUT);    //set in1PinB(12) as an output pin
  pinMode(in2PinB, OUTPUT);    //set in2PinB(14) as an output pin
  pinMode(enablePinB, OUTPUT); //set enablePinB(13) as an output pin
}


BLYNK_WRITE(V0)
{
  digitalWrite(enablePinA, param.asInt());  //run the gearhead motor A forward
  digitalWrite(in1PinA, HIGH);
  digitalWrite(in2PinA, LOW);
}
BLYNK_WRITE(V1)
{
  digitalWrite(enablePinA, param.asInt());  //run the gearhead motor A backward
  digitalWrite(in1PinA, LOW);
  digitalWrite(in2PinA, HIGH);
}
BLYNK_WRITE(V2)
{
  digitalWrite(enablePinB, param.asInt());  //run the gearhead motor B forward
  digitalWrite(in1PinB, LOW);
  digitalWrite(in2PinB, HIGH);
}


BLYNK_WRITE(V3)
{
  digitalWrite(enablePinB, param.asInt());  //run the gearhead motor B backward
 digitalWrite(in1PinB, HIGH);
  digitalWrite(in2PinB, LOW);


}


void loop() {
  Blynk.run();
}
```

8.2 Assembly Drawing