Design & Make (EG1311) Group Report

Class: B10
Submitted by: Team 1

Hwang Subin (A0255742B)
Jason Jonathan Tejaputra (A0255653Y)
Leong Wei Kang (A0253387W)
Chan Wei Ning (A0259497E)
Ian Freda Hariyanto (A0258293U)

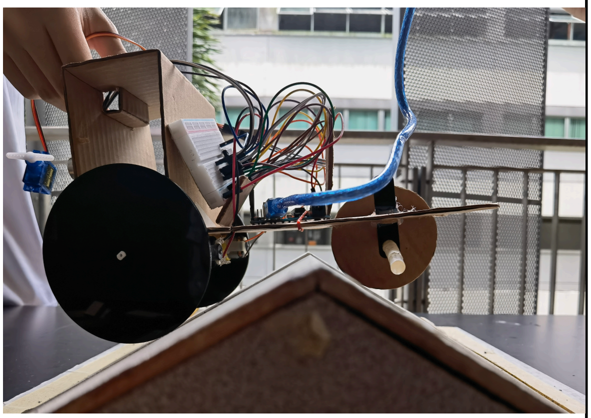
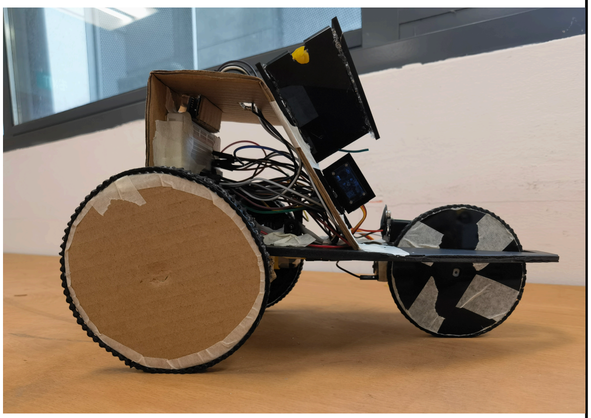
Contents:

1. Introduction

The team was tasked with designing a robot that can pass the obstacle course, carry a ping-pong ball from the starting line to the wall and propel the ball over the given wall. This report documents our journey of prototyping our vehicles, structured by its individual parts. In the following sections, we will explore our design choices, challenges we encountered, and how we overcame them. This report will hopefully embody our excitement in building the robot and the enriching experience we got.

2. Chassis

Our final model is 255mm long, 120mm wide, and 175mm high including the dimension of the wheels.

Design 1	Design 2
	
• Flat cardboard base • Rectangular shape head cardboard • Front wheels (diameter: 105mm, thickness: 3mm) • Back wheel (diameter: 85mm, thickness: 14mm) • Two motors are attached to the front wheels only • Designed without considering the catapult	• Flat foam core base • Right trapezoid shape head cardboard • Front wheels (diameter: 105mm, thickness: 10mm) • Back wheel (diameter: 85mm, thickness: 9mm) • Three motors are attached to each wheel • Catapult is attached at the back of the head

a. Initial Design

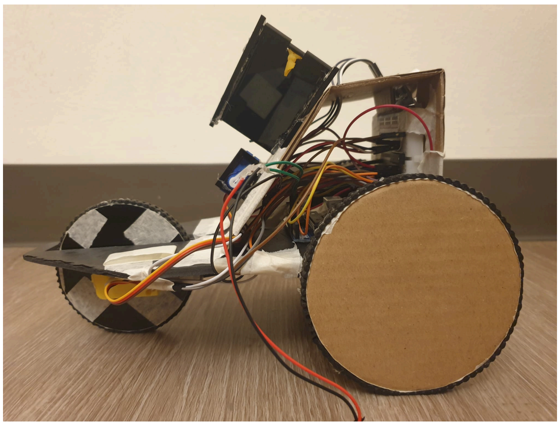
Since the design was our initial prototype, we made it all with cardboard. However, the board was not rigid enough to support all the weight of electronic materials. The base was flat and a rectangular shaped head was added by folding of the cardboard. The head raised up the sensor's position to avoid the sensor from mis-detecting the ground when the robot was going down the ramp. For the wheels, we made only 3 wheels instead of 4 wheels to minimise the weight and we made the front wheel as 105mm (diameter) and the back wheel as 85mm (diameter) referring to the measurement of the ramp and bump. We made the two front wheels out of acrylic and attached them to the motor. The back wheel was made of multi-layered cardboard and was attached to a free spinning axle. The motors were glued below the cardboard base so that the wheels could support the body of the robot. Plus, the back wheel was connected to a glue stick, which was the only material we had that was round and long, allowing the wheel to roll. However, since the glue stick was too flexible, the back wheel was staggering and disorienting. Moreover, without the motor on the back wheel, the robot did not have enough power to go over the bump.

b. Final Design

To reinforce the design, we reflected on our initial design limitations. To better support the electronics, we changed the flat base from cardboard to foam core. The rectangular head became a right-angled trapezoid in which the slope was used to place the catapult. The resulting larger space inside the head hosts the breadboard, wirings, and the Arduino Uno, which made the robot neater. To arrange the wires attached to the sensor, we made a pass-through hole in the head. To prevent the robot from staggering, we altered the thickness of the front wheels from 3mm to 10mm by attaching the cardboard to acrylic, changed the back wheel to acrylic, and added a motor to the back wheel. In addition, a non-slip mat sheet was glued around the wheels to increase the friction between the wheels and the ground.

3. Wheels

Our final design on the front wheels is two 105mm diameter with a thickness of 10mm, consisting of an acrylic piece sandwiched between 2 cardboard cutouts, with a back wheel that is 85mm with a thickness of 9mm, consisting of 3 layers of acrylic panels.



a. Surface area

On the third run we also attached 2 layers of cardboard to the front wheel to increase the surface area. This ensured that there is enough surface area for the foam mat to create suficient friction for the robot to climb through the obstacles.

b. Size and reason

For our subsequent runs, however, the back of our robot was not able to get over the bump or climb up the ramp. We concluded that more power is needed in the back wheel to pass the ramp and bump which led us to add the third motor. This was done by attaching a motor on the cardboard back wheel which led to it successfully finishing the whole obstacle course.

When doing these test runs, due to the height and size of the back wheel we also noticed that the front end of the robot got too close to the track when going down the ramp. We then decided to replace the back wheels with a smaller, acrylic wheels to ensure no toppling or damage is done to the robot as it goes up and down the ramp, in addition to ensuring that the ultrasonic sensor does not cause the robot to stop prematurely.

c. Grip materials

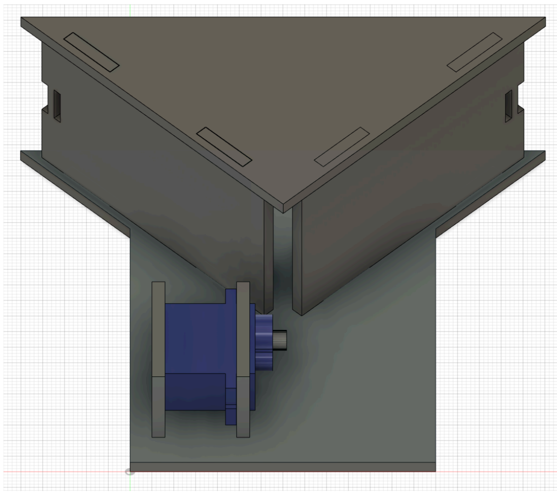
We also attached a non-slip foam mat along the circumference of all of the wheels, increasing its friction coefficient, as the initial design of the model had difficulties climbing the first obstacle.

Wheel Diameter	Wheel thickness	Additional attachment	Robot passes 1st obstacle?	Robot pass ramp?
105mm/100mm	3mm	none	Yes but very slowly	no
105mm/100mm	10mm	-foam mat	yes	Yes but very slowly
105mm/85mm	10mm	-foam mat -Cardboard attachment on the wheels	yes	yes

4. Catapult

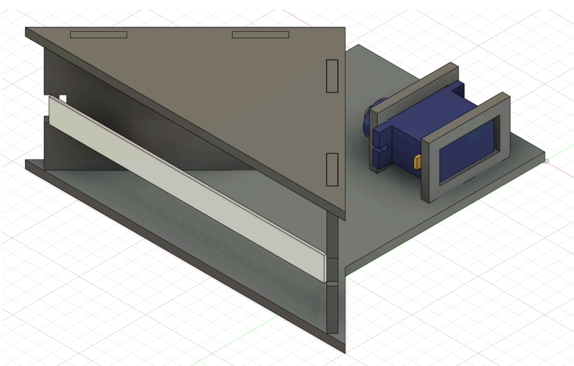
a. Design

To get the ping-pong ball over the wall, we chose to use the catapult concept, instead of the commonly used swinging arm. We chose to use acrylic and rubber band to build the catapult, with the servo motor as the trigger mechanism. Acrylic was chosen as it is rigid enough to hold the tension of the rubber band.



b. Launch Chamber

We made an enclosed triangular prism, secured using slots and matching extrusions system which is able to securely hold the ball when traversing the obstacles without any risk of the ball falling off. A thick rubber band is tied to the slots in the side plates, to act as the propellant for the ball. A small gap is left at the back to allow access to the servo.



c. Servo Trigger

The servo is secured using acrylic mounts, made to snugly fit to the exterior of the servo. A one sided servo horn is attached to the servo and used as a mounting point for the trigger. The horn is facing away from the launch chamber. A piece of spare wire is tied to the servo horn and the other end of the wire tied to the rubber band through the gap. The servo is then programmed to rotate 180° towards the chamber when the robot reaches the wall, allowing the wire to slide and release the tension in the rubber band to shoot the ball over the wall.

d. Challenges and Improvements

Upon laser cutting and assembling the pieces, we encountered some issues. Firstly, the laser cutter is more inaccurate than we anticipated. Some of the slots were significantly smaller and some were larger than the others, even though they all have the same measurements in Fusion 360. To fix this, we resorted to shaving away material using the provided pen knife for slots that are too small, and using a hot glue gun to fix the plates to slots that are too loose. As a result, the catapult was able to be assembled, but it became loose the week after likely due to the surface of the acrylic not being clean when we glue the different parts combined with the rubber band being too tight. In the end, we had to dismantle and re-glue the parts.

Secondly, mounting the wiring to the servo before and after each run was difficult as the gap is difficult to reach due to the enclosed nature of the launch chamber. Although inefficient and inelegant, we addressed this by adding one more loop to the wire piece which allows us to fish the cables through. This could be one point of improvement in the future.

Lastly, the tension of the rubber band needed to be optimised. During the first few runs, the ball launched too early as the servo failed due to the rubber band being too tight. After loosening the rubber band, the ball did not launch with enough power to go over the walls. This cycle was repeated quite a bit, as we needed to find the optimal tension on the rubber band and adjust the angle of the catapult to reliably not cause the servo to fail and also shoot the ball over the wall.

5. Coding, circuit and wiring

a. Initial design

Due to the similarities of the template project and our project, moreover for us to be more time efficient, we decided to adapt the code and circuit used in the template project for our design. Firstly, we altered the code in order for our robot to stop within 5 cm of the wall. This was done by programming our robot to stop when ‘cms < 5’ (as shown in code 1). With further testing, we realised our robot would often overshoot the stopping distance and bump into the wall. Thus, we experimented a few times and found that when ‘cms < 9’ (as shown in the final code), our robot would stop perfectly within 5cm of the wall. Hence, we used this for our final code.

In addition, to keep our robot as lightweight as possible, we decided to use 2 motors for the movement of the robot. Moreover, we used a 9V battery as it consumed less space and was lighter than using 4 AA batteries (as shown in circuit design 1).

b. Adding a servo motor for the catapult

In order to trigger the catapult that would be used to launch the ping pong ball, we added a servo motor. Initially, we set the angle of the servo motor to turn from 0 degrees to 180 degrees (as shown in code 1 and circuit design 1), this resulted in the catapult not being able to trigger as it was rotated in the wrong direction. We then changed the angle to turn from 180 degrees to 0 degrees (as shown in code 2), this made it difficult to load the wire (was initially used to load the catapult) / rubber band on the wing of the servo motor and also led to more wear and tear on the wire. Hence, we decided to reduce the angle of rotation and set the angle from 150 degrees to 0 degrees (as shown in the final code).

c. Weak connection between wire and motors

Throughout our project, we encountered many difficulties with the wiring such as the connections coming loose or getting in the way of the motion of the robot. Thus, we decided to tape and glue gun the wires to prevent it from coming loose and to reduce the clutter.

6. Final Reflections

In conclusion, our journey in making the robot has been a challenge but one that is fulfilling. As the team consists of people from different backgrounds and expertise, we have learnt what it means to work as a team. We effectively divided tasks based on our strengths and completed our parts efficiently. When issues arose, we came together and used our unique perspective to problem-solve as a collective team.

Furthermore, we displayed creativity and resourcefulness in designing our robots. One example is using the glue stick as the wheel axle. Although this did not work out in the end, we reflected on this failure which inspired our final three-wheeled design. This experience teaches us the importance of brainstorming and thinking outside the box.

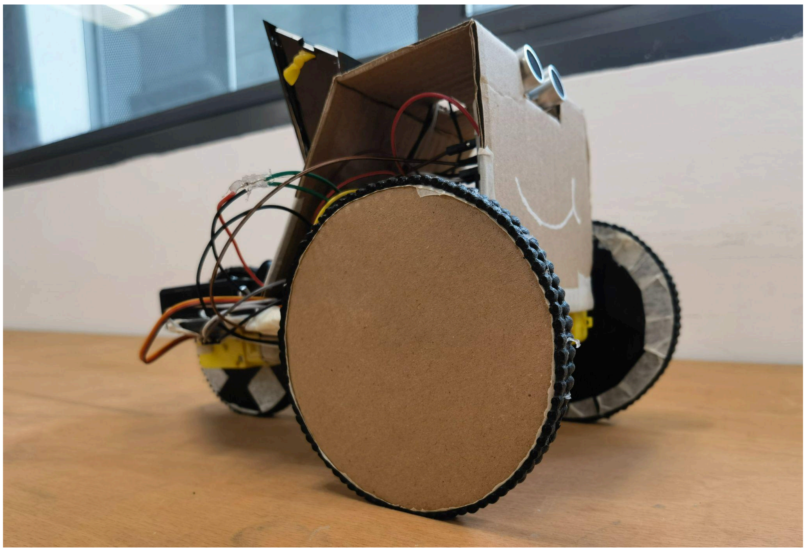
In the beginning, we were unsure of how to build the robot. However, the professor and TAs were incredibly supportive in pushing us to prototype and create our first design, regardless of its performance. Prof Jason Ku’s lecture regarding prototyping and the TED Talk video he showed made us realise how crucial it is to get started, test our prototypes and learn from our failures instead of thinking about how to achieve perfection from the very start.

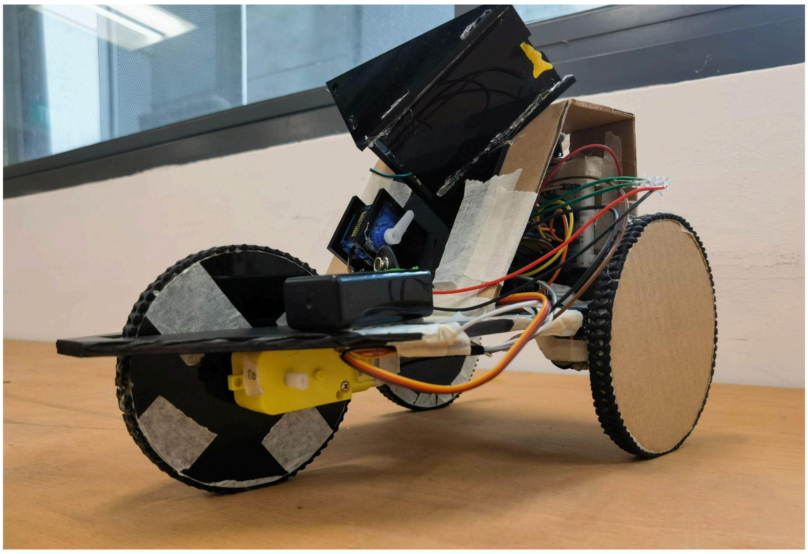
The above is proven true because many unexpected problems occurred during our runs on the obstacle course. A problem even occurred during one of the actual test runs. Although this initially caught us off guard, the countless challenges we experienced during our design process have prepared us well. We were doing our best to stay calm and be flexible in adapting to challenges because we knew that if we persevered, we could collectively find a solution. In the end, our robot overcame the obstacle course.

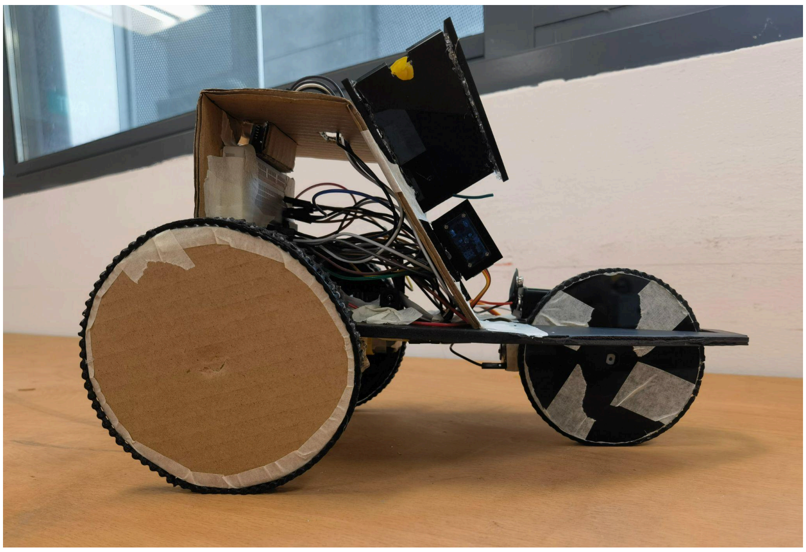
Hence, to close this report. We like to express our gratitude to the Professors, TAs and lab assistants who have been incredibly helpful. Having you with us has made our learning journey more meaningful and enjoyable. Thank you!

7. Appendix

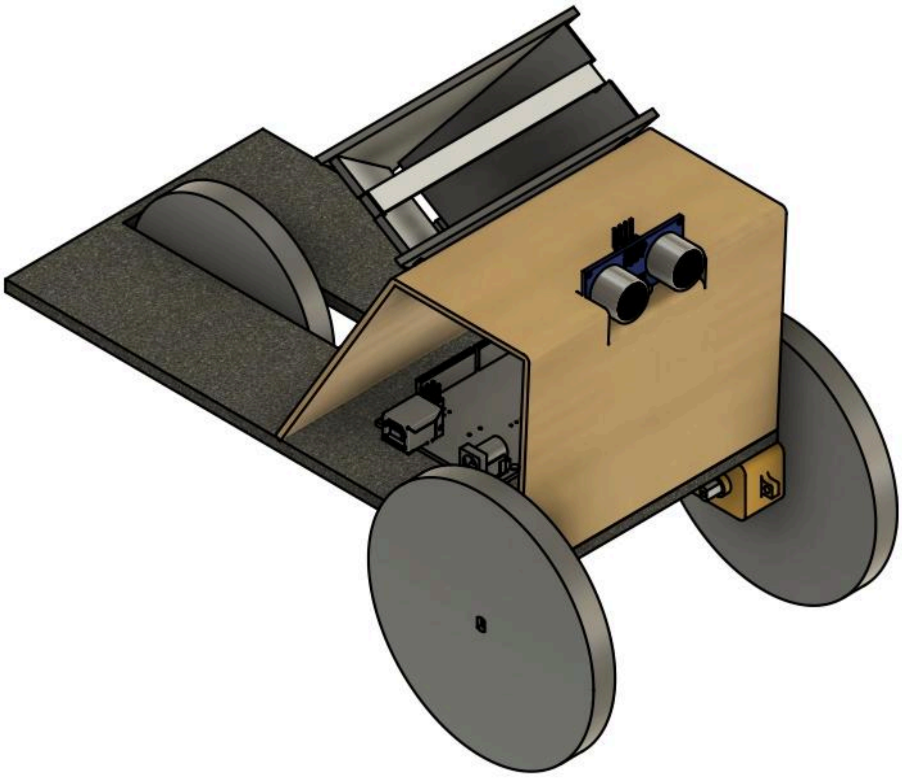
A. Photograph of Final Robot

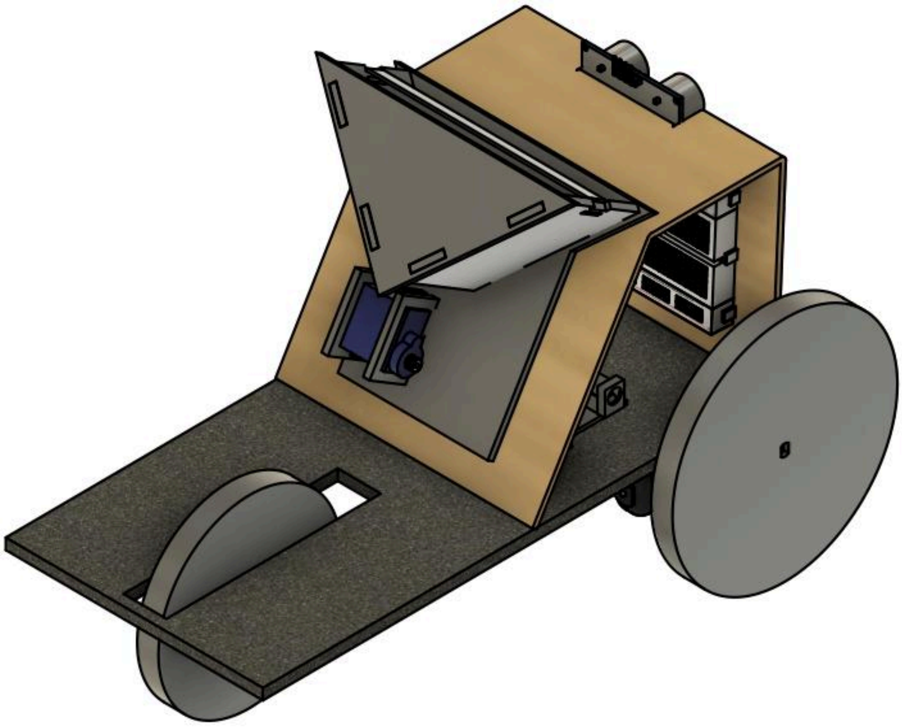




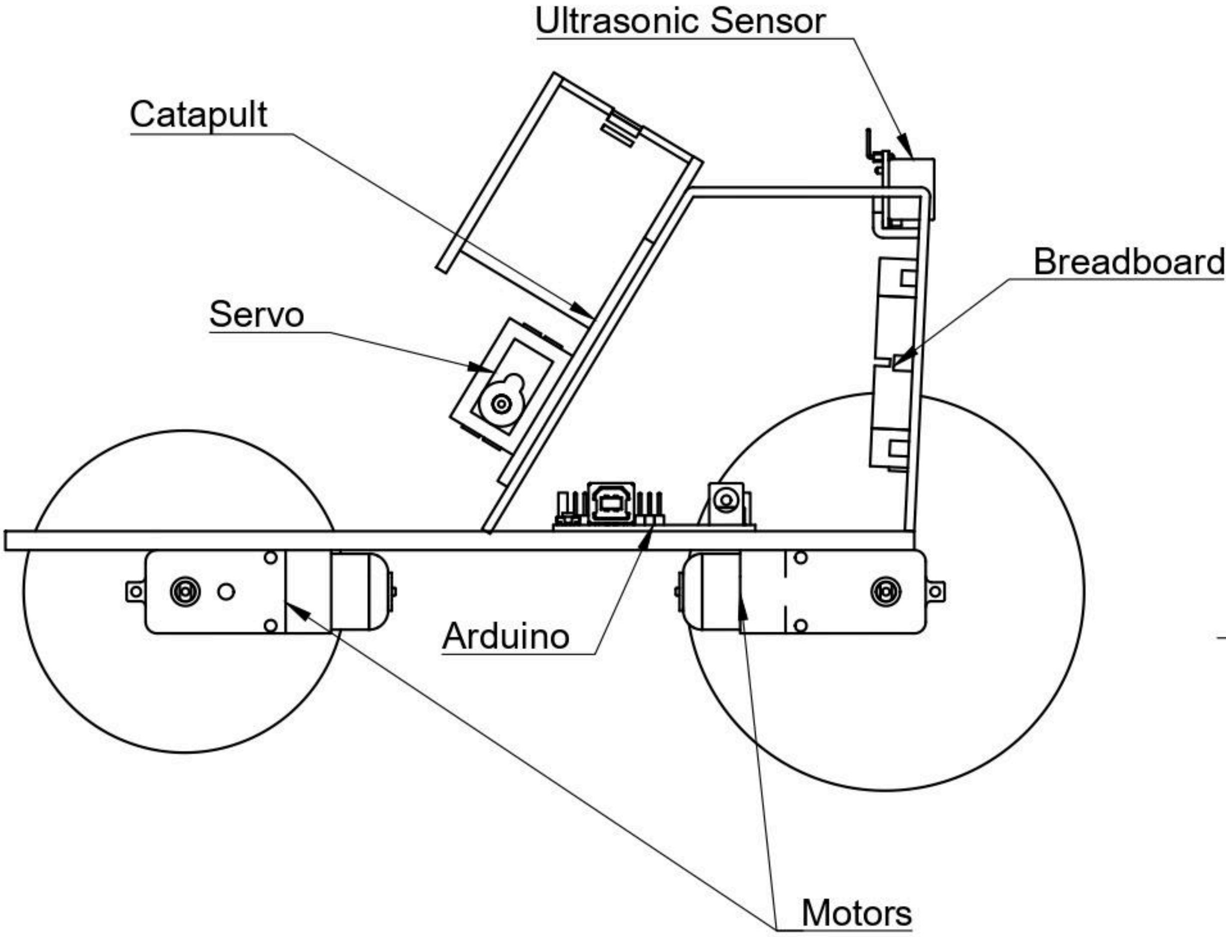


B. CAD rendering & Drawings

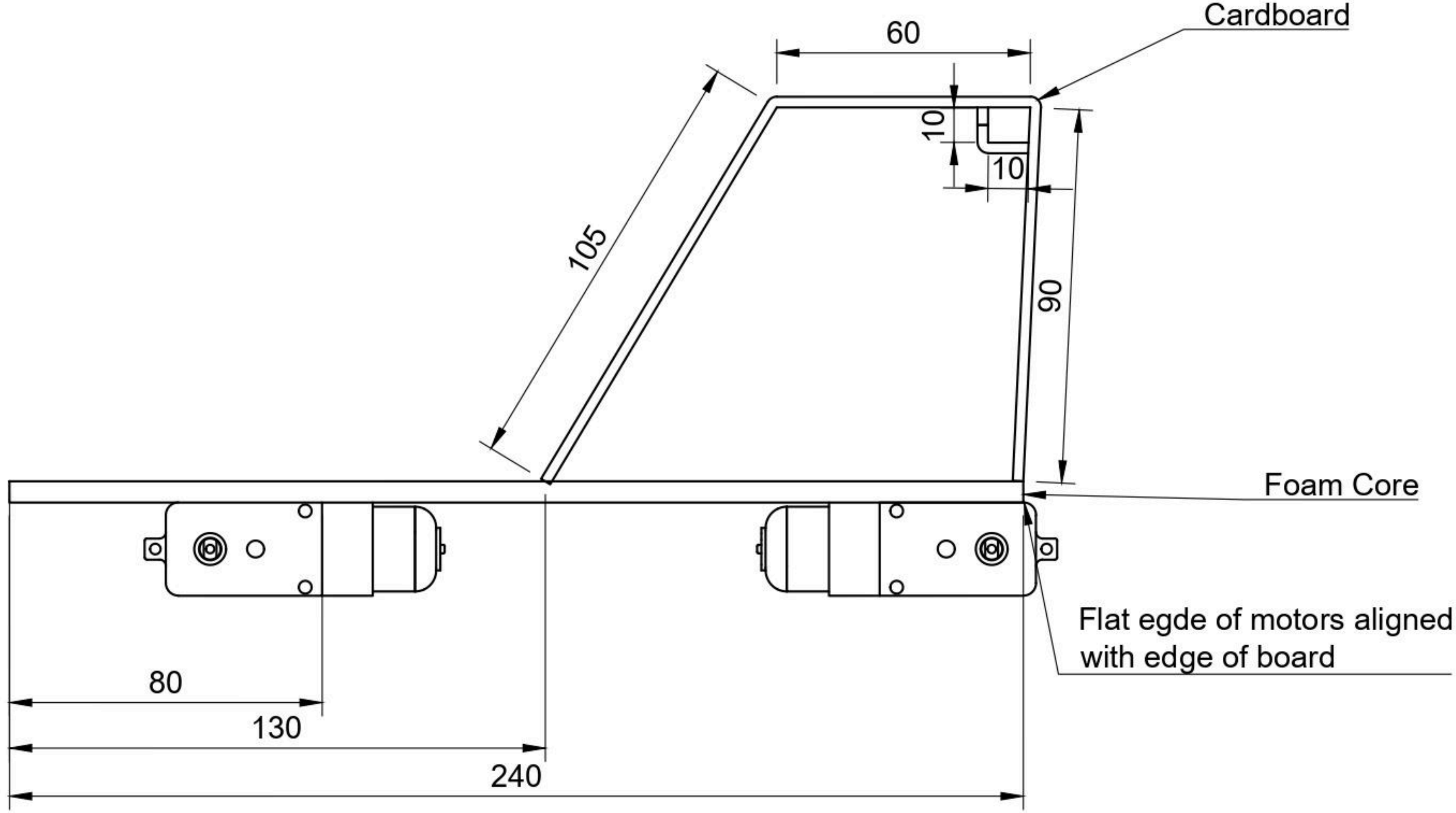




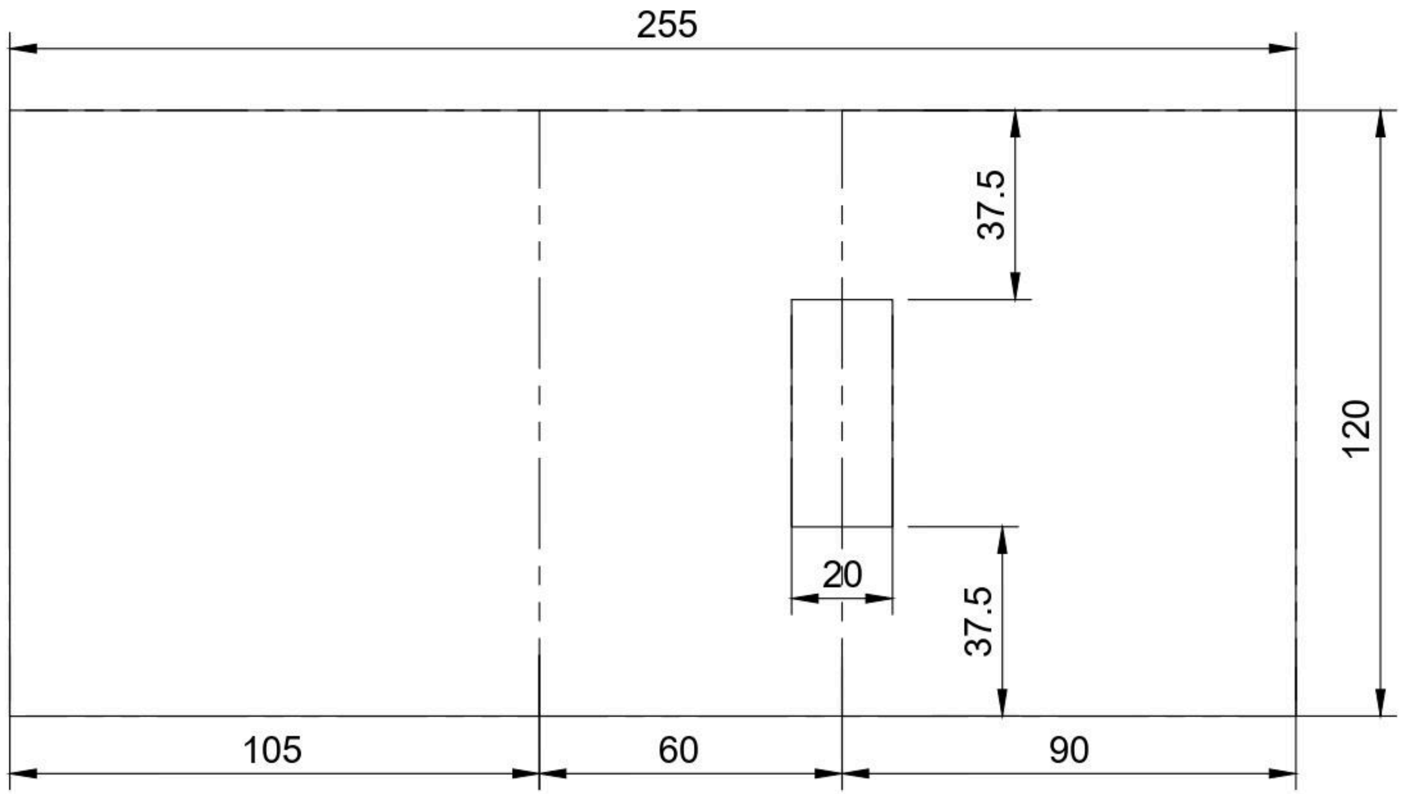
Dept.	Technical reference 1:2.5	Created by B10 Group 1 10/28/2022	Approved by
		Document type	Document status
		Title Overview	DWG No.
			Rev. | Date of issue | Sheet 1/7



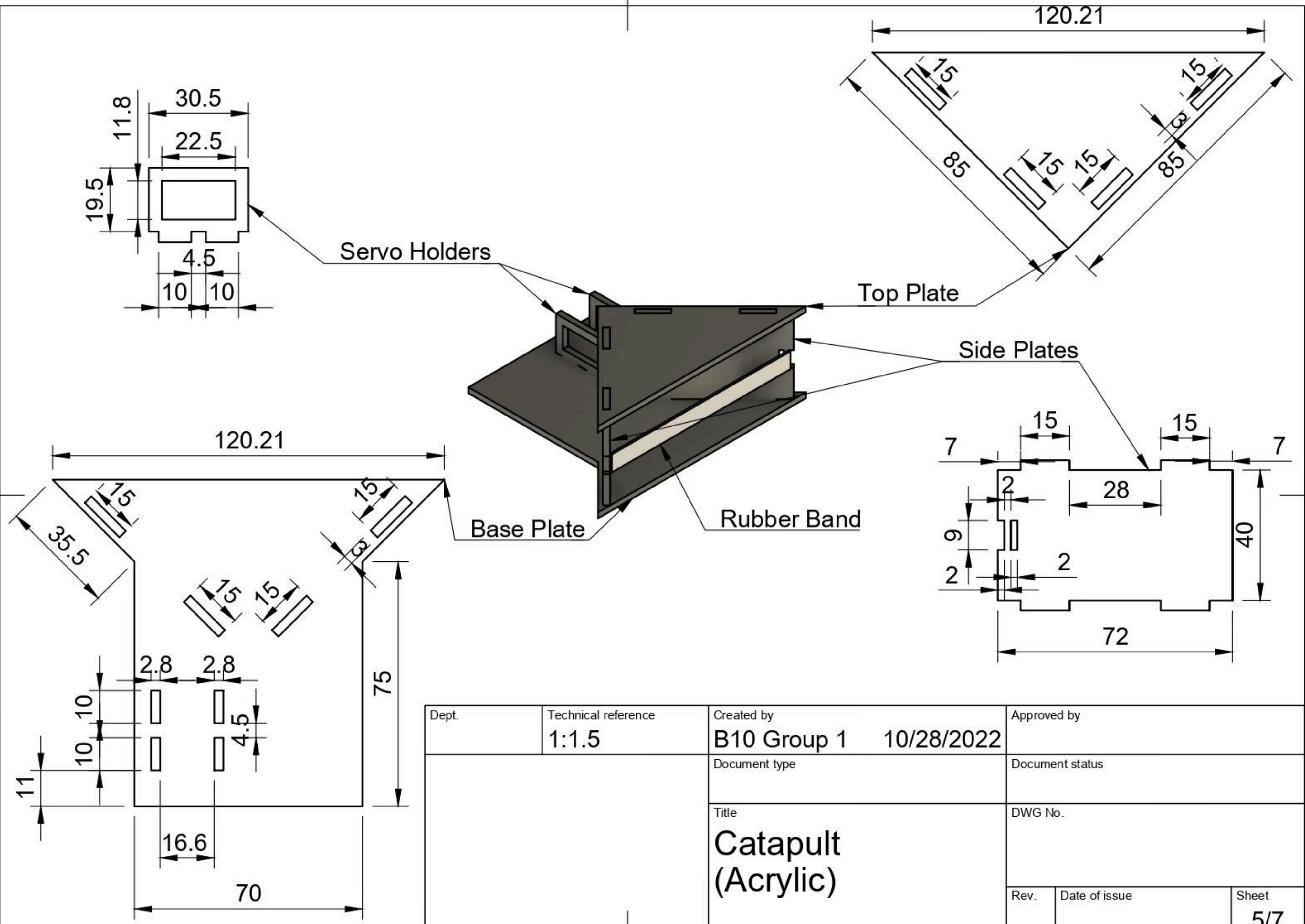
Dept.	Technical reference	Created by		Approved by		
	1:2	B10 Group 1	10/28/2022			
		Document type		Document status		
		Title		DWG No.		
		Side Component View (Front Right Wheel Removed)		Rev.	Date of issue	Sheet 2/7



Dept.	Technical reference	Created by	Approved by
	1:1.5	B10 Group 1 10/28/2022	
		Document type	Document status
		Title: Frame (Foam & Cardboard)	DWG No.
			Rev. / Date of issue / Sheet 3/7

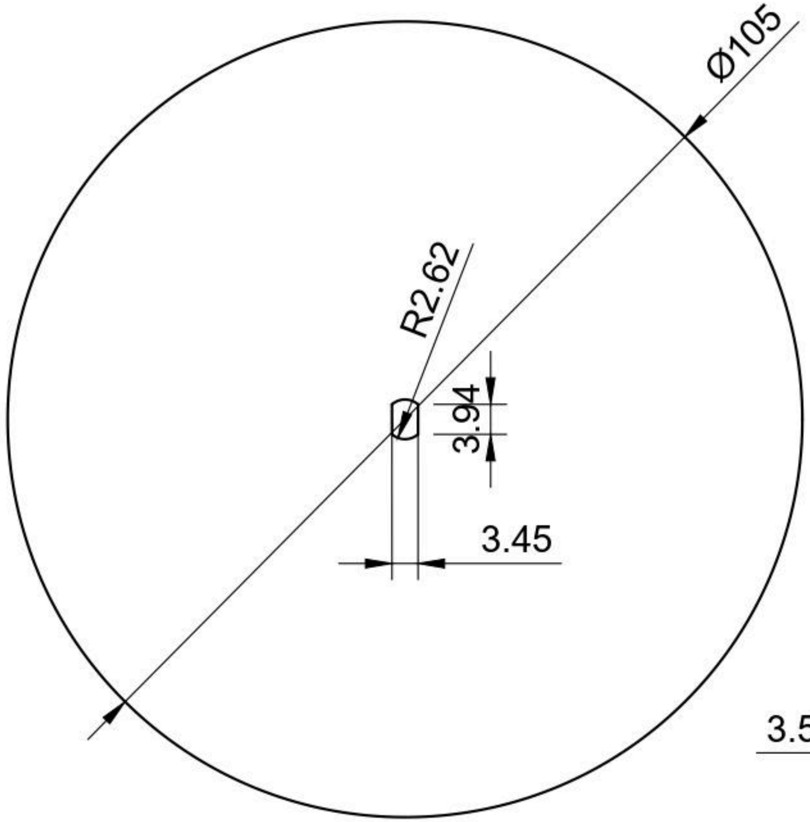


Dept.	Technical reference	Created by	Approved by
	1:1.5	B10 Group 1 10/28/2022	
		Document type	Document status
		Title: Head Cover Template (Cardboard)	DWG No.
			Rev.	Date of issue	Sheet 4/7

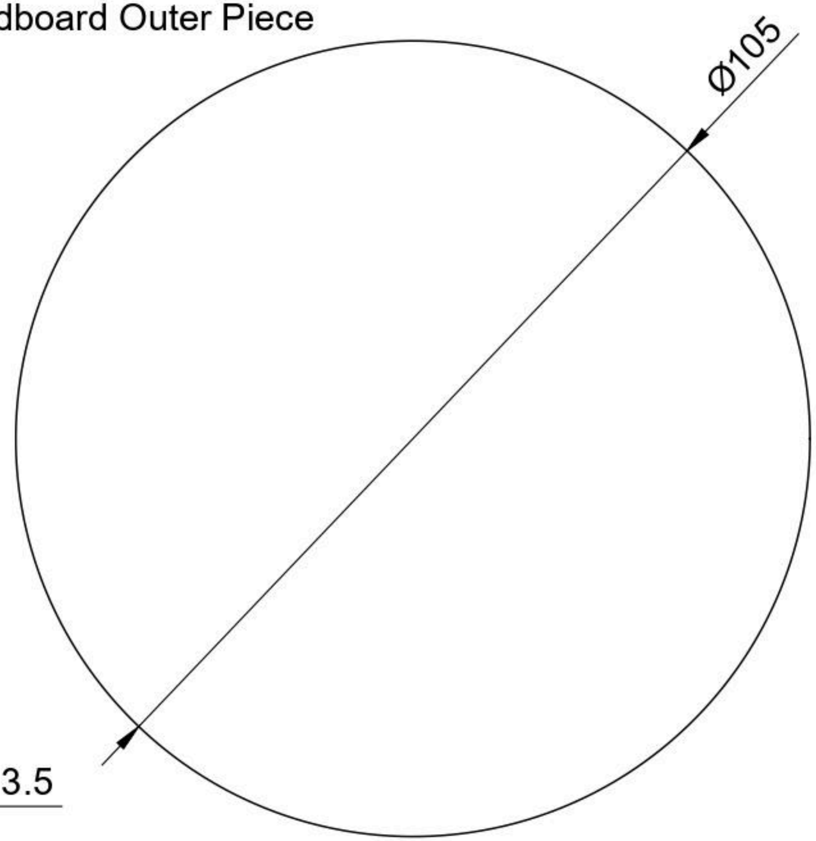


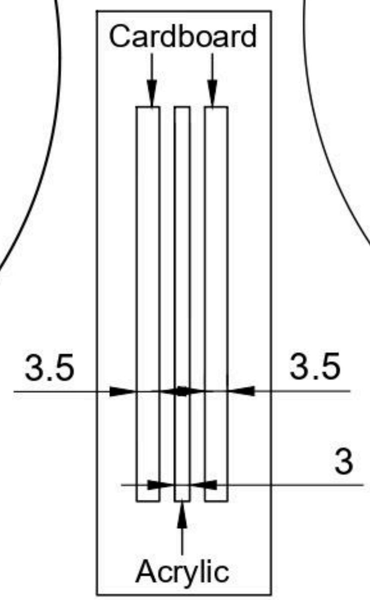
Dept.	Technical reference	Created by	Approved by
	1:1.5	B10 Group 1 10/28/2022	
		Document type	Document status
		Title Catapult (Acrylic)	DWG No.
			Rev. / Date of issue / Sheet 5/7

Acrylic Inner Piece



Cardboard Outer Piece

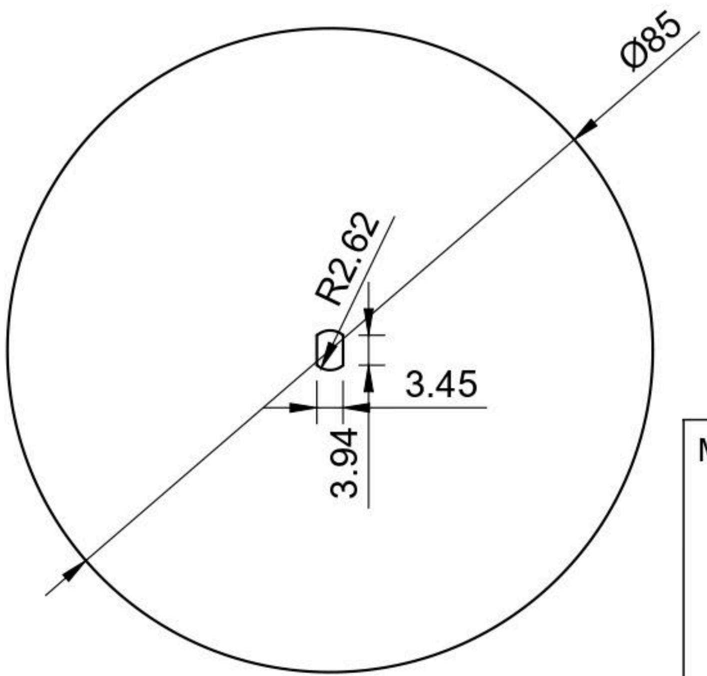




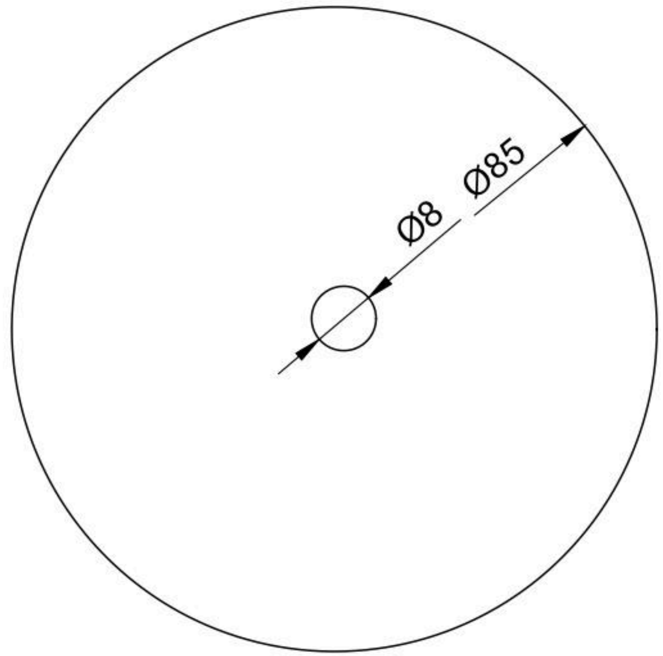
Note:
All wheels are wrapped
along their circumference
using non-slip mat strips

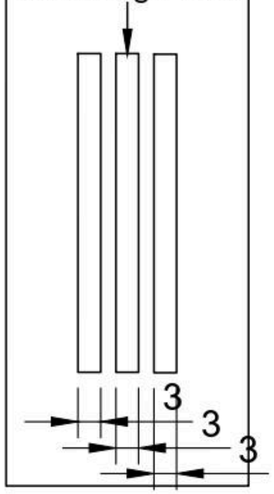
Dept.	Technical reference	Created by	Approved by		
	1:1	B10 Group 1 10/28/2022			
		Document type	Document status		
		Title Front Wheels (Cardboard & Acrylic)	DWG No.		
			Rev.	Date of issue	Sheet 6/7

Inner Mounting Piece



Outer Piece

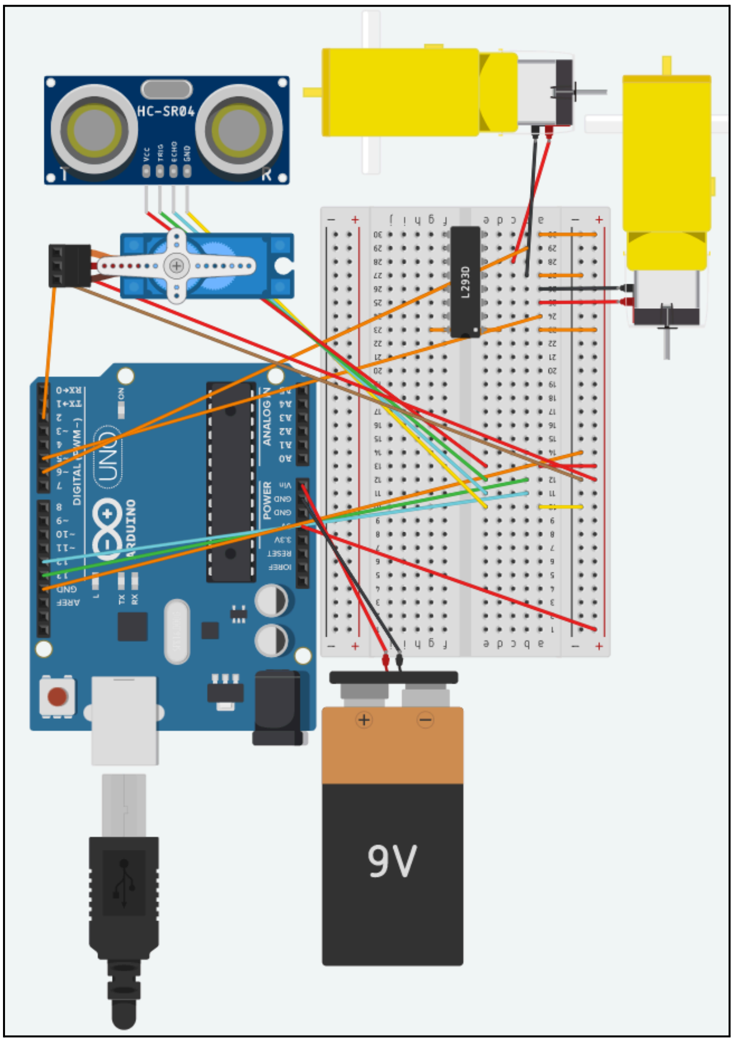


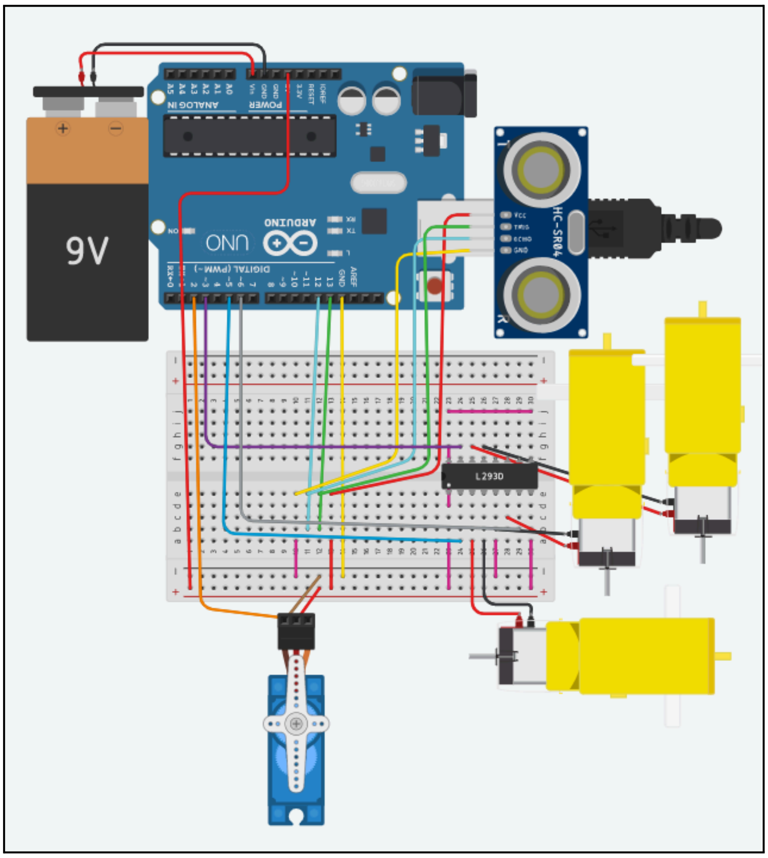


Note:
All wheels are wrapped
along their circumference
using non-slip mat strips

Dept.	Technical reference	Created by	Approved by
	1:1	B10 Group 1 10/28/2022	
		Document type	Document status
		Title: Rear Wheel (Acrylic)	DWG No.
			Rev. / Date of issue / Sheet 7/7

C. Circuit Design





Final Circuit Design

Initial Circuit Design

D. Arduino Code

```
//
#include <Servo.h>
Servo servo;
 int servo_pin = 2;
 int TRIG_PIN = 13;
 int ECHO_PIN = 12;
 int MOTOR_PIN1 = 6;
 int MOTOR_PIN2 = 5;
 float SPEED_OF_SOUND = 0.0345;

void setup() {
 pinMode(MOTOR_PIN1,OUTPUT);
 pinMode(MOTOR_PIN2,OUTPUT);
 pinMode(TRIG_PIN,OUTPUT);
 digitalWrite(TRIG_PIN,LOW);
 pinMode(ECHO_PIN, INPUT);
 Serial.begin(9600);
 servo.attach(servo_pin, 660, 2400);

}

void loop() {
 digitalWrite(TRIG_PIN,HIGH);
 delayMicroseconds(10);
 digitalWrite(TRIG_PIN,LOW);
 int microsecs = pulseIn(ECHO_PIN,HIGH);
 float cms = microsecs * SPEED_OF_SOUND/2;
 Serial.println(cms);
 if (cms<5){
  digitalWrite(MOTOR_PIN1,LOW);
  digitalWrite(MOTOR_PIN2,LOW);
  servo.write(0);
  servo.write(180);
 } else{
  digitalWrite(MOTOR_PIN1,HIGH);
  digitalWrite(MOTOR_PIN2,HIGH);
  servo.write(0);
 }
  delay(10);
}
```

Code 1

```
#include <Servo.h>
Servo servo;
 int servo_pin = 2;
 int TRIG_PIN = 13;
 int ECHO_PIN = 12;
 int MOTOR_PIN1 = 6;
 int MOTOR_PIN2 = 5;
 int MOTOR_PIN3 = 3;
 float SPEED_OF_SOUND = 0.0345;

void setup() {
 pinMode(MOTOR_PIN1,OUTPUT);
 pinMode(MOTOR_PIN2,OUTPUT);
 pinMode(MOTOR_PIN3,OUTPUT);
 pinMode(TRIG_PIN,OUTPUT);
 digitalWrite(TRIG_PIN,LOW);
 pinMode(ECHO_PIN, INPUT);
 Serial.begin(9600);
 servo.attach(servo_pin, 660, 2400);

}

void loop() {
 digitalWrite(TRIG_PIN,HIGH);
 delayMicroseconds(10);
 digitalWrite(TRIG_PIN,LOW);
 int microsecs = pulseIn(ECHO_PIN,HIGH);
 float cms = microsecs * SPEED_OF_SOUND/2;
 Serial.println(cms);
 if (cms<10){
  digitalWrite(MOTOR_PIN1,LOW);
  digitalWrite(MOTOR_PIN2,LOW);
  digitalWrite(MOTOR_PIN3,LOW);
  servo.write(0);
  servo.write(180);
 } else{
  digitalWrite(MOTOR_PIN1,HIGH);
  digitalWrite(MOTOR_PIN2,HIGH);
  digitalWrite(MOTOR_PIN3,HIGH);

  servo.write(0);
 }
  delay(10);
}
```

Code 2

```
#include <Servo.h>
Servo servo;
 int servo_pin = 2;
 int TRIG_PIN = 13;
 int ECHO_PIN = 12;
 int MOTOR_PIN1 = 6;
 int MOTOR_PIN2 = 5;
 int MOTOR_PIN3 = 3;
 float SPEED_OF_SOUND = 0.0345;

void setup() {
 pinMode(MOTOR_PIN1,OUTPUT);
 pinMode(MOTOR_PIN2,OUTPUT);
 pinMode(MOTOR_PIN3,OUTPUT);
 pinMode(TRIG_PIN,OUTPUT);
 digitalWrite(TRIG_PIN,LOW);
 pinMode(ECHO_PIN, INPUT);
 Serial.begin(9600);
 servo.attach(servo_pin, 660, 2400);

}

void loop() {
 digitalWrite(TRIG_PIN,HIGH);
 delayMicroseconds(10);
 digitalWrite(TRIG_PIN,LOW);
 int microsecs = pulseIn(ECHO_PIN,HIGH);
 float cms = microsecs * SPEED_OF_SOUND/2;
 Serial.println(cms);
 if (cms<9){
  digitalWrite(MOTOR_PIN1,LOW);
  digitalWrite(MOTOR_PIN2,LOW);
  digitalWrite(MOTOR_PIN3,LOW);
  servo.write(150);
  servo.write(0);
 } else{
  digitalWrite(MOTOR_PIN1,HIGH);
  digitalWrite(MOTOR_PIN2,HIGH);
  digitalWrite(MOTOR_PIN3,HIGH);

  servo.write(150);
 }
  delay(10);
}
```

Final Code