Technical Design (TD)

Project: Autonomous Electric Car

Company: BlitzDrive

Place, date:	Enschede, 20th March 2024
Drawn up by:	Aritra Biswas
	Mathieu Geiller
	Jimmy Tithphit
	Nghia Hoang
	Dung Phan
Version:	0.1

Place, date:	Saxion Hogeschool, Enschede 15-03-2024		
Drawn by:	BlitzDrive		
Chapters 2, 4, 6	Aritra Biswas	545606	545606@student.saxion.nl
Chapters 1, 4, 6	Mathieu Geiller	553016	553016@student.saxion.nl
Chapters 2, 3, 4, 6	Nghia Hoang	553429	553429@student.saxion.nl
Chapters 3, 4, 6	Jimmy Tithphit	518275	518275@student.saxion.nl
Chapters 2, 4, 5	Dung Phan	546821	546821@student.saxion.nl
Version Number:	[0.1]: The first draft of the technical design of our project.		
	[1.0]:		
Clients:	Saxion Hogeschool		

Table of Contents

I. List of figures 5
II. Abbreviations 5
1 Introduction 1
1.1 High-level design results 1
Simulation For Ultrasonic sensor: 1
Simulation for Tilt Sensor: 4
Simulation for IR sensor: 8
2 Analysis of requirements 10
3 Component/Unit/Module Selection 13
3.1 Options 13
Microcontroller 13
Tilt Sensor 13
Obstacle Detection 14
Line Detection 14
Wheels 15
Motors 16
Battery 16
3.2 Comparison 17
3.3 List of selected components/units/modules 18
4 Technical design overview 19
4.1 Overview of initial hardware design 19
4.2 Interfaces 20
5 Technical requirements 22
5.1.1 Component #1 - Battery 22
5.1.2 Component #2 - Motor 22
5.1.3 Component #3 - IR Sensor 22
5.1.4 Component #4 - Tilt Sensor 22
5.1.5 Component #5 - Ultrasonic Sensor 23
5.1.6 Component #6 - Arduino Uno 23
5.1.7 Software function #1 (Line Detection) 23
5.1.8 Software function #2 (RampControl) 24
5.1.9 Software function #3 (ObstacleDetection) 24
6 Detailed technical designs 25
6.1 Mechanical 25
6.2 Electrical 26
6.3 Electronic 27
6.4 Software 28
References 30

I. List of figures

Figure 1.1 Simulation Design........1
Figure 1.2 Simulation Running........2
Figure 1.3 Build for Figure 1.2........3
Figure 1.4 Results of Simulation........4
Figure 1.5 Build for Figure 1.4........4
Figure 1.6 Simulating........5
Figure 1.7 Tilting the Tilt........6
Figure 1.8 Not Tilting the Tilt........7
Figure 1.9 Build for Figure 1.8........8
Figure 1.10 Result........9
Figure 4.1: High-level design with specific components........20
Figure 4.2 Board Topology........21
Figure 4.3 Voltage Regulator........22
Figure 4.4 Resistor........22
Figure 6.1 Car Model 3D........26
Figure 6.2 Altium Design........27
Figure 6.3 PCB Design........28

II. Abbreviations

DSP	Digital Signal Processing
FFT	Fast Fourier Transform
TL;DR	Too long; didn't read

1 Introduction

This is the Technical Design Document for BlitzDrive, this document will assist clients and other prospects with our product and guide them through our process of the car.

Our aims and objectives for the project are:

- To provide a technical overview of the car on the components we are using and how they work or how they are connected.
- Simulations of the circuit in TinkerCAD to understand the components.
- Calculating and selecting components.
- Making a PCB design with the selected components from the High-level design.

1.1 High-level design results

Simulation For Ultrasonic sensor:

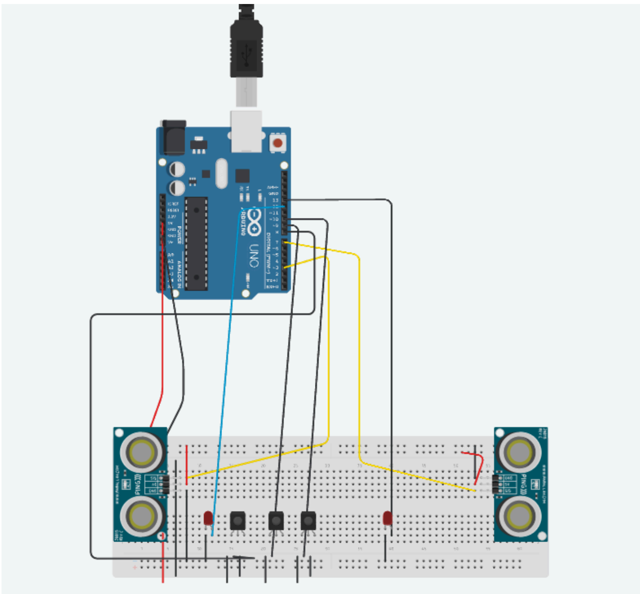


Figure 1.1 Simulation Design

Here is the simulation that we did for the IR sensors and Ultrasonic sensors. To test this we used this code:

```
C/C++

  if (distance1 < 50 || distance2 < 50) {
    digitalWrite(ledPin, HIGH);
  } else {
    digitalWrite(ledPin, LOW);
  }

  delay(100);
}
```

In this code, we are saying if we detect anything within the 50 cm range then the LED will light up (as seen under Figure 1.2), else it will stay off. This helps to see whether the ultrasonic sensor works when an object is placed in front of it.

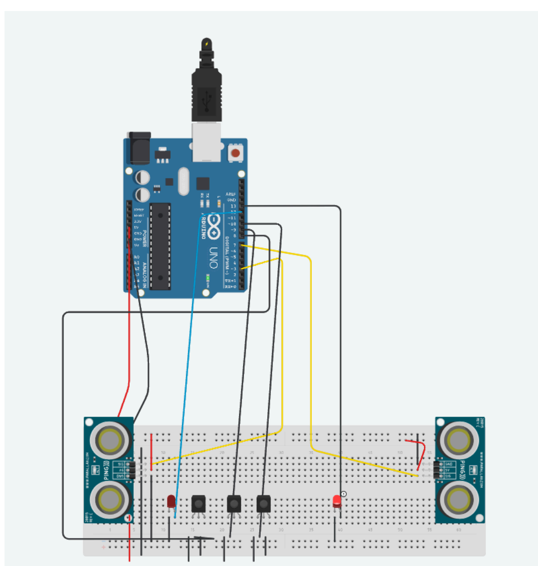


Figure 1.2 Simulation Running

Figure 1.2 is the result of the ultrasonic, as we can see, the LED lights up when an1T object is placed in front of it.

Figure 1.3 is the proof of concept for the ultrasonic:

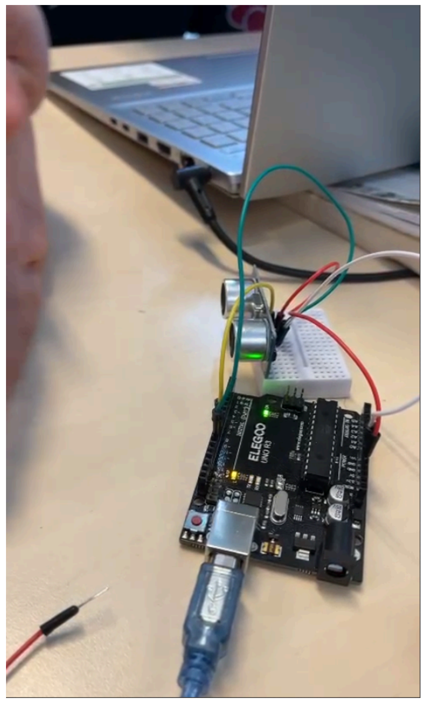


Figure 1.3 Build for Figure 1.2

This Ultrasonic sensor is different from the one of Tinkercad as the real one has 4 pins, one pin for ground, another for the power, and the trigger and echo pins that are connected to the Arduino. In the test, one of our group members placed their hand in front of the sensor and moved it back and forth to see the distance measured (check Figure 1.4).

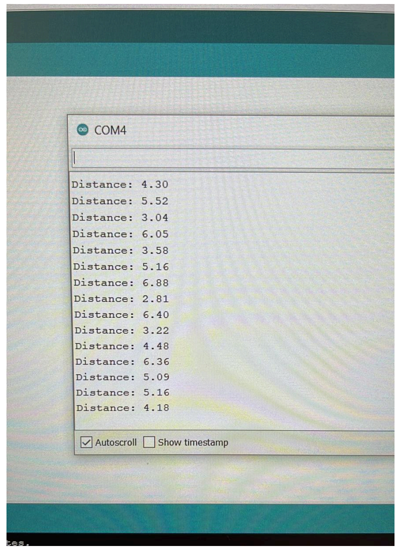


Figure 1.4 Results of Simulation

Simulation for Tilt Sensor:

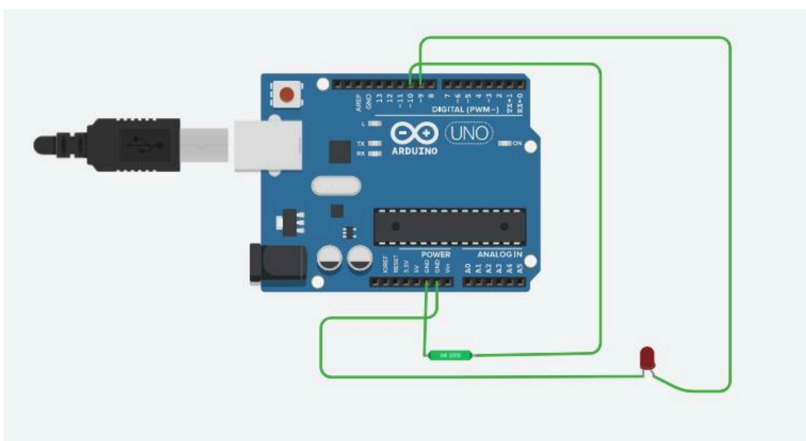


Figure 1.5 Build for Figure 1.4

We are now testing the tilt sensor, one pin connected to the Arduino and another to the 5V.

```
C/C++
void setup() {
  pinMode(10, INPUT_PULLUP);
  pinMode(9, OUTPUT);
  Serial.begin(9600);
}

void loop() {
  int number = digitalRead(10);
  Serial.println(number);

  if (number == 1)digitalWrite(9, HIGH);
  else digitalWrite(9, LOW);
}
```

In the code we define pin 10 as the input pin, so the pin for the tilt sensor, and pin 9 as the output pin, the LED pin. Then we loop saying the int number is equal to whatever we read from the tilt sensor, and then print the number. Then if an incline is detected the LED is HIGH, so turned on if not it is turned off.

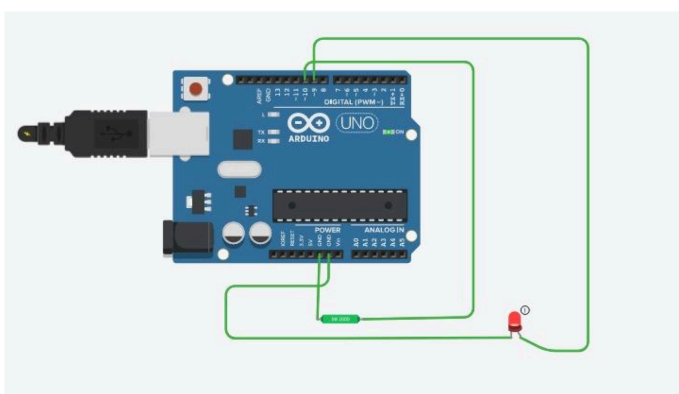


Figure 1.6 Simulating

Here is the proof of concept for the tilt sensor:

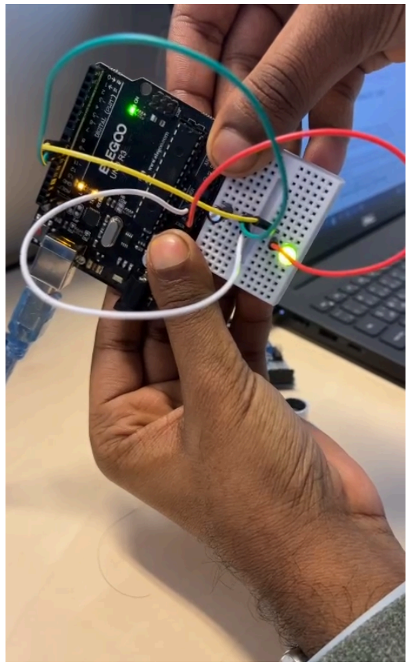


Figure 1.7 Tilting the Tilt

We tested the Tilt sensor by tilting the breadboard and if the tilt sensor detects an incline, then the LED will light up (as seen in Figure 1.7) and if no incline is detected then the LED shouldn't light up, (as seen in Figure 1.8).

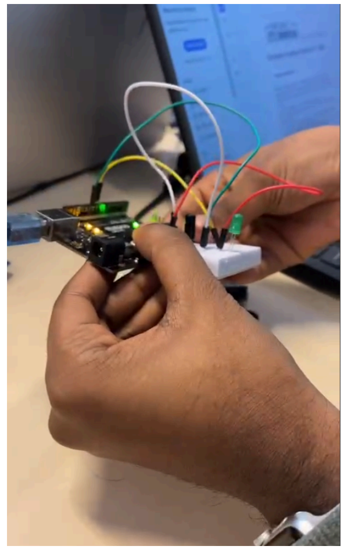


Figure 1.8 Not Tilting the Tilt

Here is the code we used to test the tilt sensor:

```
C/C++
void setup() {
  pinMode(10, INPUT_PULLUP);
  pinMode(9, OUTPUT);
  Serial.begin(9600);
}

void loop() {
  int smthn = digitalRead(10);
  Serial.println(smthn);

  if (smthn == 1)digitalWrite(9, HIGH);
  else digitalWrite(9, LOW);
}
```

Similar code that we used on Tinkercad to test the tilt sensor first in a simulation.

Simulation for IR sensor:

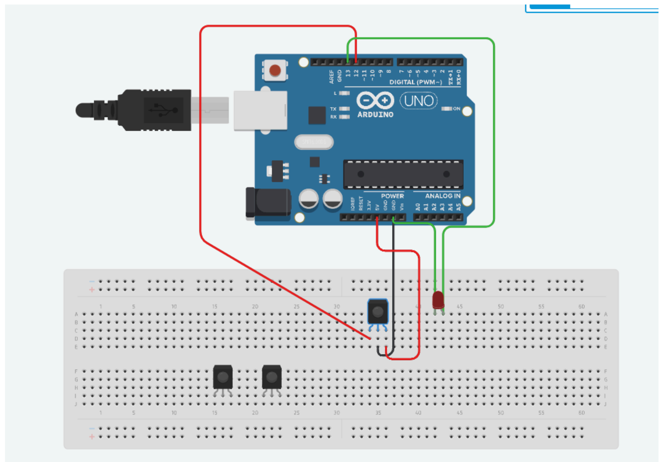


Figure 1.9 Build for Figure 1.8

Here we are testing the IR sensor, if it detects something the LED will light up.

```
C/C++
const int irSensorPin = 12;
const int ledPin = 13;

void setup() {
    pinMode(irSensorPin, INPUT);
    pinMode(ledPin, OUTPUT);
}

void loop() {
    bool read = digitalRead(irSensorPin);

    // Check if IR sensor detects an object
    if (read == HIGH) {
        digitalWrite(ledPin, HIGH);
    } else {
        digitalWrite(ledPin, LOW);
    }

    delay(100);
}
```

In the code, we are first designating which pins are the IR sensor and the LED. And then setting up which pin is going to be an input (IR sensor) and which pin is going to be the output (LED). After that we read what we are getting from the IR sensor, and if something is detected the LED is HIGH meaning it is on (as we can see in Figure 1.10), if not it will stay off.

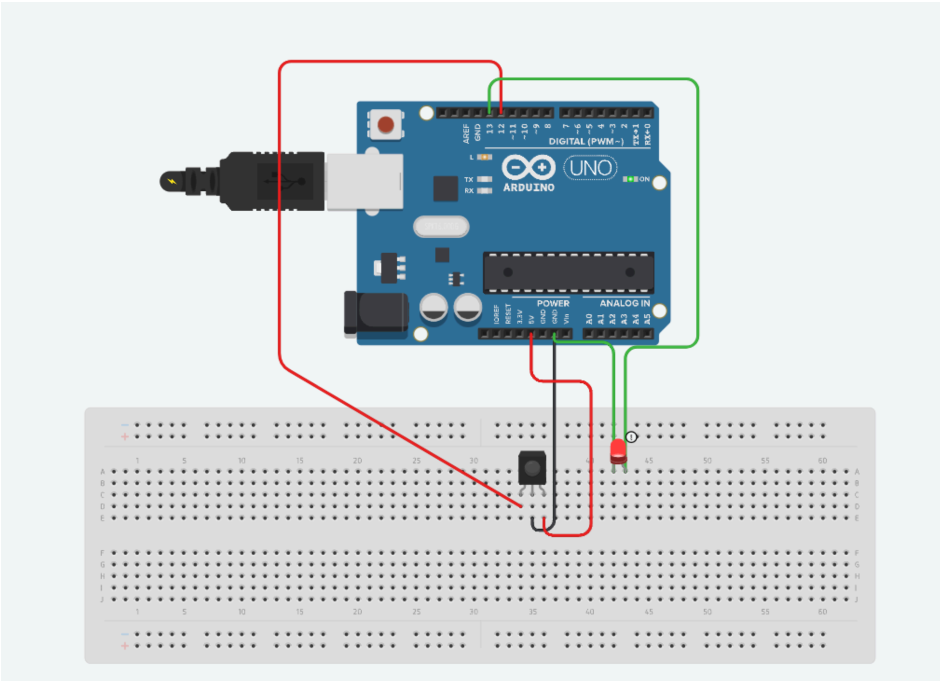


Figure 1.10 Result

2 Analysis of requirements

Microcontroller	Processing Power	Memory	Price	Power Consumption	Compatibility	Final Rating
Arduino UNO	20%	20%	30%	10%	20%	Good for use

Why was Arduino UNO chosen?

The largest technical criteria for choosing an ideal microcontroller for this project lies in the processing power of the microcontroller. There were strong contenders like the Raspberry Pi for this criteria which has a good processing power (by clock speed) but was not ideal in terms of price. [1] Since our budget was limited to 50 EUR, the Arduino Uno was the most ideal option. The Raspberry Pi, though better in terms of memory, was over-budget and filled with features requiring higher technical expertise such as Linux knowledge. [2][3]

Tilt Sensor	Accuracy	Sensitivity	Range	Price	Response time	Final Rating
KY-020 Tilt Sensor	30%	10%	35%	15%	15%	Good for use

Why was the KY-020 Tilt Sensor chosen?

The KY-020 is chosen since it fits all required criteria. It is cheap [5] and accurate since it is a simple closed-circuit sensor. It is chosen over other tilt switches like KY-017 since sensitivity is not the biggest criterion in this case.[4]

Obstacle Detection Sensor	Detection Range	Accuracy	Response Time	Field of View	Price	Final Rating
HC-SR04 Ultrasonic Sensor	20%	10%	30%	10%	30%	Good for use

Why was the HC-SR04 chosen?

The HC-SR04 was chosen mostly because of the availability and experience the team had working with it. [6]

Line Detection Sensor	Accuracy	Detection Range	Response Time	Resolution	Price	Grades
Azdelivery HW 201	40%	5%	15%	10%	30%	Good for use

Why was the HW - 201 IR Sensor Chosen?

The duality between accuracy and price is taken into the highest consideration while choosing a significant IR sensor to detect a line. The HW-201 works on the principle of reflection of light reflection from the surroundings to the receiver built into the component. It's the ideal choice since it has high accuracy and a favourable 35-degree angle detection range. [7]

Motors	Power Rating	Torque	Compatibility	Efficiency	Price	Final Rating
Thrust Eco 15	20%	15%	25%	25%	15%	Good for use

Why was the Thrust Eco 15 chosen?

The brand of the motor was not the deciding factor but rather the Torque rating for the voltage provided was the factor taken into consideration for the operation of the car instead. Most motors have similar specifications and the real distinction comes between brushless and brushed DC motors. In this case, a brushed DC motor (of brand Thrust Eco 15) was chosen. Since efficiency held a higher requirement standard than price, a brushless motor was chosen.[8]

Battery	Power Rating	Capacity	Size and Weight	Dimensions	Efficiency	Final Rating
Conrad Energy NiMH Battery	25%	30%	10%	10%	25%	Good for use

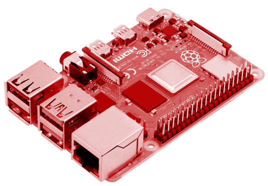
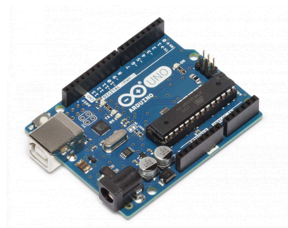
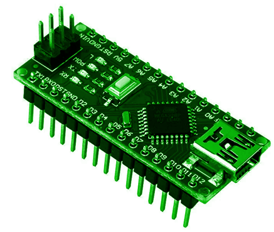
Why was the Conrad Energy NiMH Battery chosen?

The NiMH batteries have a higher energy density compared to other rechargeable batteries, allowing them to store more energy per unit volume. This attribute makes them suitable for powering portable electronic devices. Besides, NiMH batteries are less harmful to the environment than traditional disposable batteries.[9]

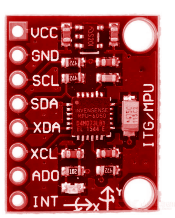
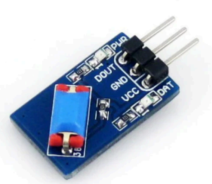
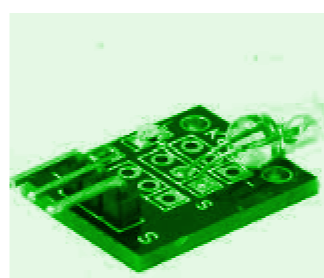
3 Component/Unit/Module Selection

3.1 Options

Microcontroller

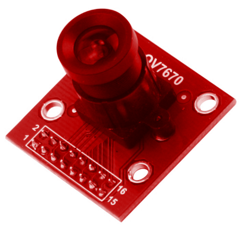
		
Raspberry Pi [10]	Arduino Uno[10]	Arduino Nano[11]
Advantages: • Processing Power • Connectivity • Versatility (Can be use in many projects) Disadvantages: • Expensive • Power consumption • Too complex • Not open source	Advantages: • Simplicity • Open source • Budget-friendly • Less power consumption Disadvantages: • Limited capability • doesn't support wireless connectivity	Advantages: • Easy to use • small and compact • cheapest among all the options • fit on the breadboard itself Disadvantages: • Very limited capability • doesn't support wireless connectivity •

Tilt Sensor

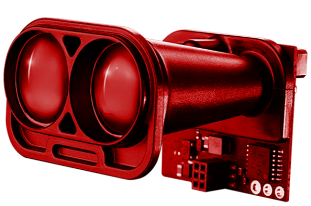
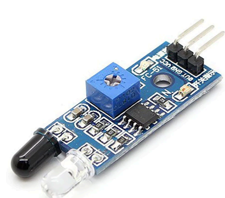
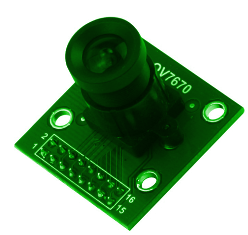
		
MPU 6050 Gyro [12]	KY 020 [13]	KY - 017 [14] [15]
Advantages: • Provides accurate and reliable	Advantages: • Less expensive than alternative	Advantages: • Less expensive than alternative

measurements • Fast in operation • Small and lightweight, can be easily fit on car Disadvantages: • More expensive than alternative applications • Subjected to relative drift	applications • Not subjected to relative drift Disadvantages: • Less sensitivity	applications • Not subjected to relative drift • More sensitivity Disadvantages: • Environmentally unfriendly • Outdated

Obstacle Detection

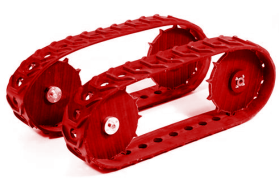
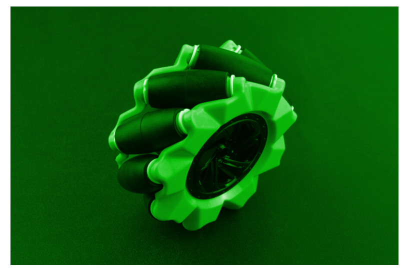
		
Camera Sensor [16]	Ultrasonic Sensor [17]	LiDar Sensor [18]
Advantages: • It can detect a wide range of obstacles such as (Human, animal, car, etc. Disadvantages: • Weather conditions will affect the usage of the sensor • Expensive	Advantages: • Budget friendly • Reflect sound so conditional environment won't affect the result Disadvantages: • accuracy • limitation range • won't work in a vacuum where sound can't travel around	Advantages: • Accuracy • Speed Disadvantages: • Complexity • Expensive

Line Detection

		
LiDar Sensor[18]	IR Sensor[19]	Camera Sensor[16]
Advantages:	Advantages:	Advantages:

• Accuracy • Speed Disadvantages: • Complexity • Expensive	• Low power consumption • Good reaction time Disadvantages: • Frequency affected by smoke, dust, haze, and daylight • Can control only one gadget at a time	• It can detect a wide range of obstacles such as (Human, animal, car, etc. Disadvantages: • Weather conditions will affect the usage of the sensor • Expensive

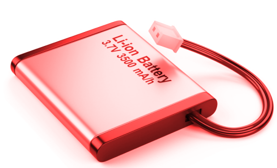
Wheels

		
Track Wheels[20]	Wheels[20]	Mecanum wheels
Advantages: • Traction • weight growth potential Disadvantages: • Speed	Advantages: • Budget-friendly • speed • simplicity Disadvantages: • Wear out easily • Traction	Advantages: • Traction • Potential • Can manoeuvre sideways Disadvantages: • Speed • Weight

Motors

		
Thrust Eco 15 (Brushed motor) [21]	RC Car Motor 4300KV 900W 50000 RPM Low Noise High Torque Brushless Motor [22]	4050 Brushless Motor 4-Pole 3000kv/3Y [22]
Advantages: ● Low cost ● Reliable ● Simplicity ● Can handle rough environment Disadvantages: ● Short lifespan ● Have issue controlling high speed	Advantages: ● Higher torque ● High energy efficiency ● Compatible Disadvantages: ● Complexity ● Expensive	Advantages: ● Less maintenance required ● Long operating life Disadvantages: ● Complexity ● Expensive

Battery

		
Nickel-based Battery [23]	Lithium-ion Battery [24]	Alkaline Battery [25]

Advantages: ● Delivers high current output. ● Relatively tolerant of overcharging. Disadvantages: ● Environmentally unfriendly ● Little tolerance for overcharging	Advantages: ● Compatible ● Long lifespan Disadvantages: ● Expensive ● Easily burned	Advantages: ● Ease of use ● Cheap Disadvantages: ● Short lifespan ● Less compatible

3.2 Comparison

	Speed	Weight	Accuracy	Power Usage	Budget
	10%	15%	35%	20%	20%
Concept 1	++	- -	++	--	- -
Concept 2	-	++	+	+	+
Concept 3	+	+	+	++	++

[+] = Compatible

[++] = Very Compatible

[-] = Bad

[- -] = Very Bad

3.3 List of selected components/units/modules

Component	Name
Microcontroller	Arduino Uno R3
IR Sensor	Adafruit 5
Ultrasound Sensor	HC-SR04
Motor	Thrust Eco 15
Battery	Conrad 7.2V

4 Technical design overview

4.1 Overview of initial hardware design

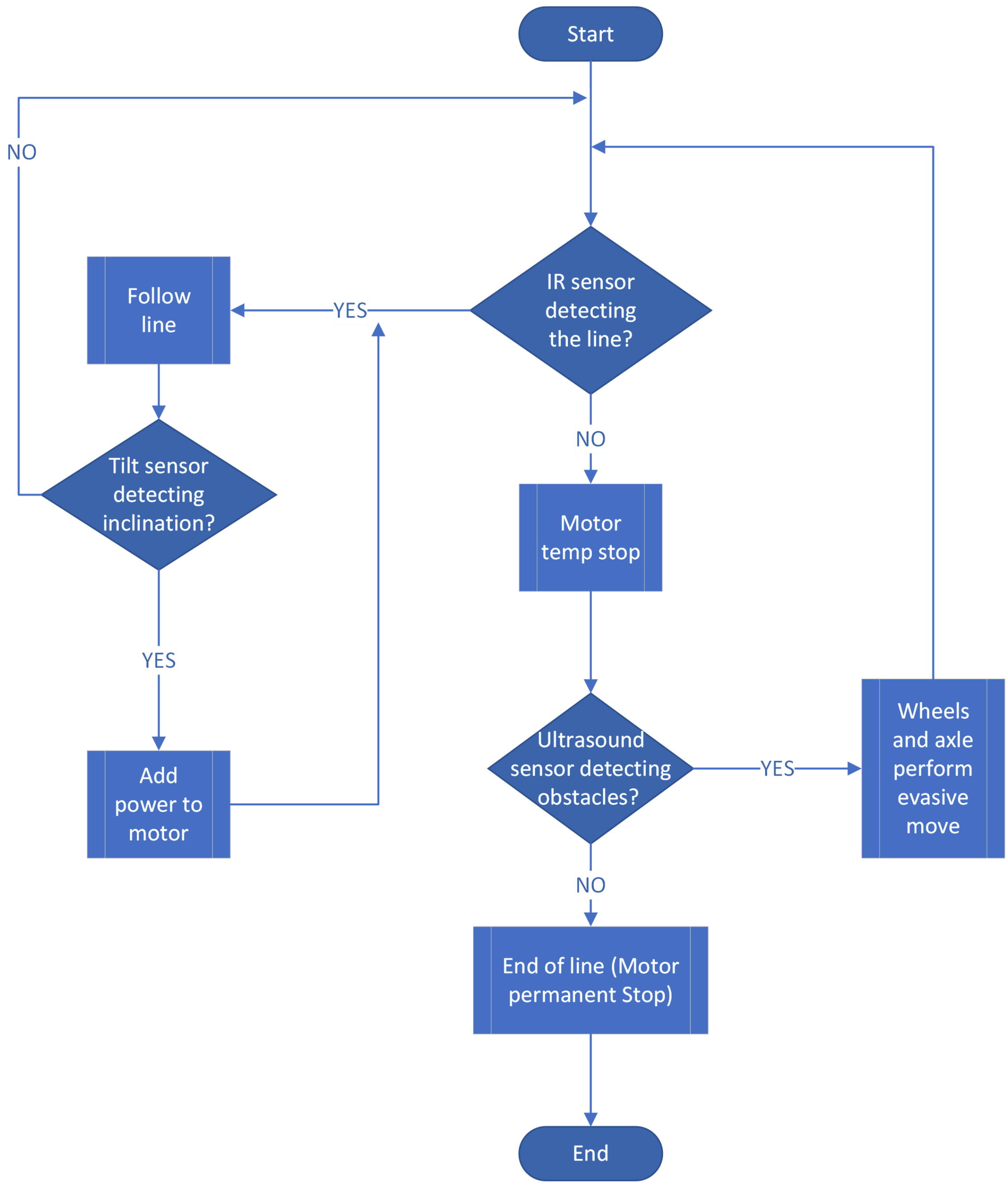


Figure 4.1: High-level design with specific components

Figure 4.1 illustrates the high-level flowchart with all the selected components that our team has discussed. We decided to use IR Sensors to detect the line and stop at the endpoint of the race. Besides, we use the Tilt Sensor to check an inclination. If there is an inclination, the speed of the motor will be accelerated. Finally, we use Ultrasonic Sensors to detect the object. When the object is detected, at a specific distance, the car will make a manoeuvre move to avoid the obstacle, then find the line and keep following that line.

4.2 Interfaces

In the Appendix A, the system design is included from the Functional Design document for reference. It is the overview of the model that is followed while planning the technical design.

Not all components can handle the 12V that the battery provides, for example, the ultrasonic sensors, IR sensors, or Tilt sensor. To prevent our components from getting short-circuited, we can use voltage regulators.

Voltage Regulator 1:

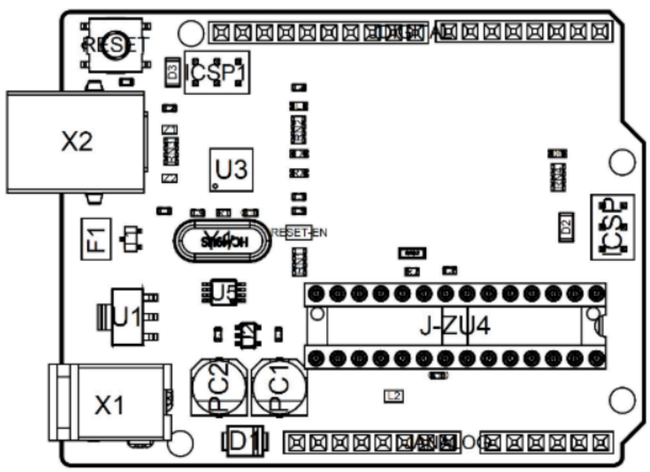


Figure 4.2 Board Topology

Here is the board topology for the Arduino Uno R3, as we can see, the U1 is the voltage regulator that is pre-installed on the board. We can use that by connecting it to X1 and using the voltage regulator.[3]

Voltage Regulator 2:

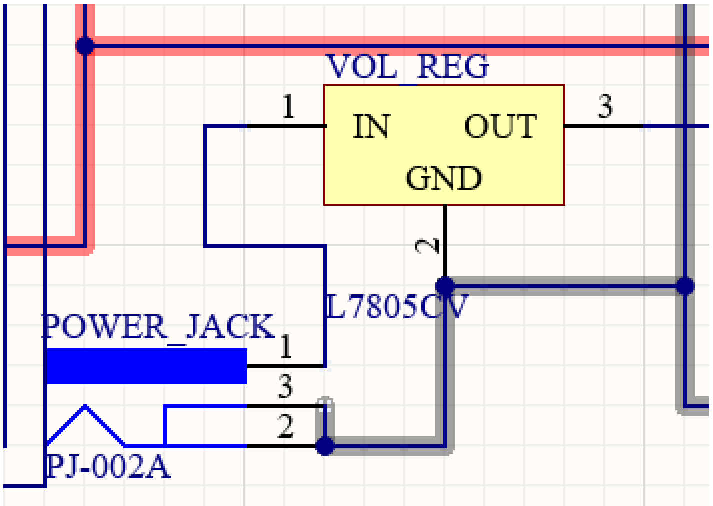


Figure 4.3 Voltage Regulator

Here is the voltage regulator from Altium, it is connected to the power jack, so where the 12V comes from and the in goes into the voltage regulator and comes out 5V or 3V depending on which components.

Resistors:

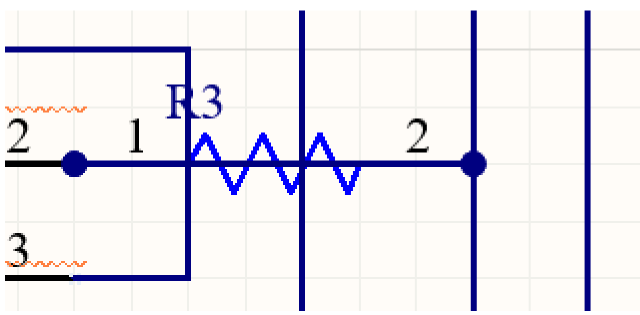


Figure 4.4 Resistor

We also use 0 ohms resistors in Altium as we can then put any resistance we want and test it. Because putting any other resistance might not work so we chose a neutral resistance to keep our options open or if we make calculation errors. Hence, 0 ohms resistors are added as a redundancy.

5 Technical requirements

5.1.1 Component #1 - Battery

NiHM Battery - Conrad Energy 7.2 V 2000 Mah[26]

#	Requirement	Relation	Value	Unit
TR001	Maximum voltage level	≤	7.2	Volt
TR002	Number of cycles	>	1000	Cycles
TR003	Complete discharge time for a single charge at maximum power	>	30	minutes
TR004	Choose the right charger	≤	10	Ampere

5.1.2 Component #2 - Motor

Absima Thrust Standard 540 Size Brushed Motor - 2310060[27]

#	Requirement	Relation	Value	Unit
TR005	Nominal/operating voltage level	=	7.2	Volt
TR006	Maximum current level	≤	16	Amps
TR007	Revolution per minute (RPM)	=	32.000	RMP

5.1.3 Component #3 - IR Sensor

IR Sensor - AZ-Delivery[27]

#	Requirement	Relation	Value	Unit
TR008	Nominal/operating voltage level	=	3.3	Volt
TR009	Minimum detection range	=	2	cm
TR010	Maxium detection range	=	30	cm

5.1.4 Component #4 - Tilt Sensor

Tilt Sensor - KY-020[13]

#	Requirement	Relation	Value	Unit
TR011	Nominal/operating voltage level	=	5	Volts

TR012	Number of axles	=	1	Axis

5.1.5 Component #5 - Ultrasonic Sensor

Ultrasonic Sensor - HC-SR04[6]

#	Requirement	Relation	Value	Unit
TR013	Nominal/operating voltage level	=	5	Volt
TR014	Maximum current level	≤	2	mA
TR015	Measuring angle	=	30	degree
TR016	Resolution	=	0.3	cm
TR017	Minimum detection range	=	2	cm
TR018	Maxium detection range	=	400	cm

5.1.6 Component #6 - Arduino Uno

Arduino Uno - Microcontroller[3]

# [3]	Requirement	Relation	Value	Unit
TR019	Nominal/operating voltage level	=	5	Volt
TR020	Minimum Input Voltage (recommended)	>	7	Volt
TR021	Maxium Input Voltage (recommended)	<	12	Volt
TR022	Minimum Input Voltage (limit)	>	6	Volt
TR023	Maxium Input Voltage (limit)	<	20	Volt

5.1.7 Software function #1 (Line Detection)

#	Requirement
TR101	When ValueSensorOne = 0, ValueSensorTwo = 0, ValueSensorThree = 0, then FindTheLine()
TR102	When ValueSensorOne = 0, ValueSensorTwo = 1, ValueSensorThree = 0, then FollowTheLine()

#	Requirement
TR103	When ValueSensorOne = 1, ValueSensorTwo = 1, ValueSensorThree = 0, then SteeringLeft()
TR104	When ValueSensorOne = 0, ValueSensorTwo = 1, ValueSensorThree = 1, then SteeringRight()
TR105	When ValueSensorOne = 1, ValueSensorTwo = 1, ValueSensorThree = 1, then Stop()

5.1.8 Software function #2 (RampControl)

#	Requirement
TR106	When ValueSensor = HIGH, then PowerMotorIncrease()
TR107	When ValueSensor = LOW, then PowerMotorDecrease()

5.1.9 Software function #3 (ObstacleDetection)

#	Requirement
TR108	When ObstacleDetection=TRUE, then PowerMotorDecrease()
TR109	When ValueSensorLeft = 0, ValueSensorFront > 15, ValueSensorRight > 15, then SteeringLeft()
TR109	When ValueSensorLeft >15, ValueSensorFront > 15, ValueSensorRight = 0, then SteeringRight()
TR110	When ValueSensorFront = 0, then FollowTheLine()

6 Detailed technical designs

6.1 Mechanical

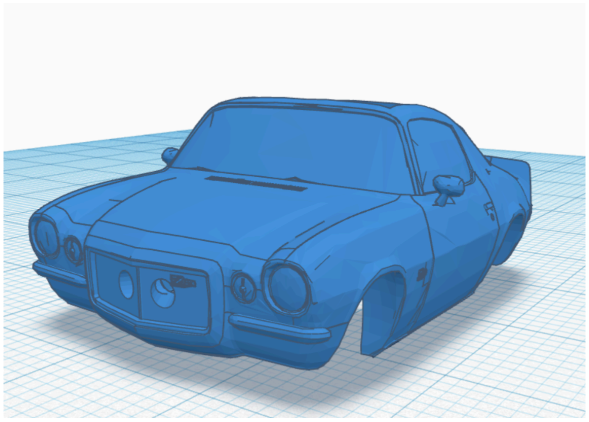


Figure 6.1 Car Model 3D

Dimensions of the car body:

- Wheels: 85mm
- Body Length: 340mm
- Body Width: 260mm
- Car height: 150mm

The design (Figure 6.1) was made on "TinkerCAD". The dimensions are made by measuring the components we got from the tutors and ensuring we have another space to fit all the components inside.

6.2 Electrical

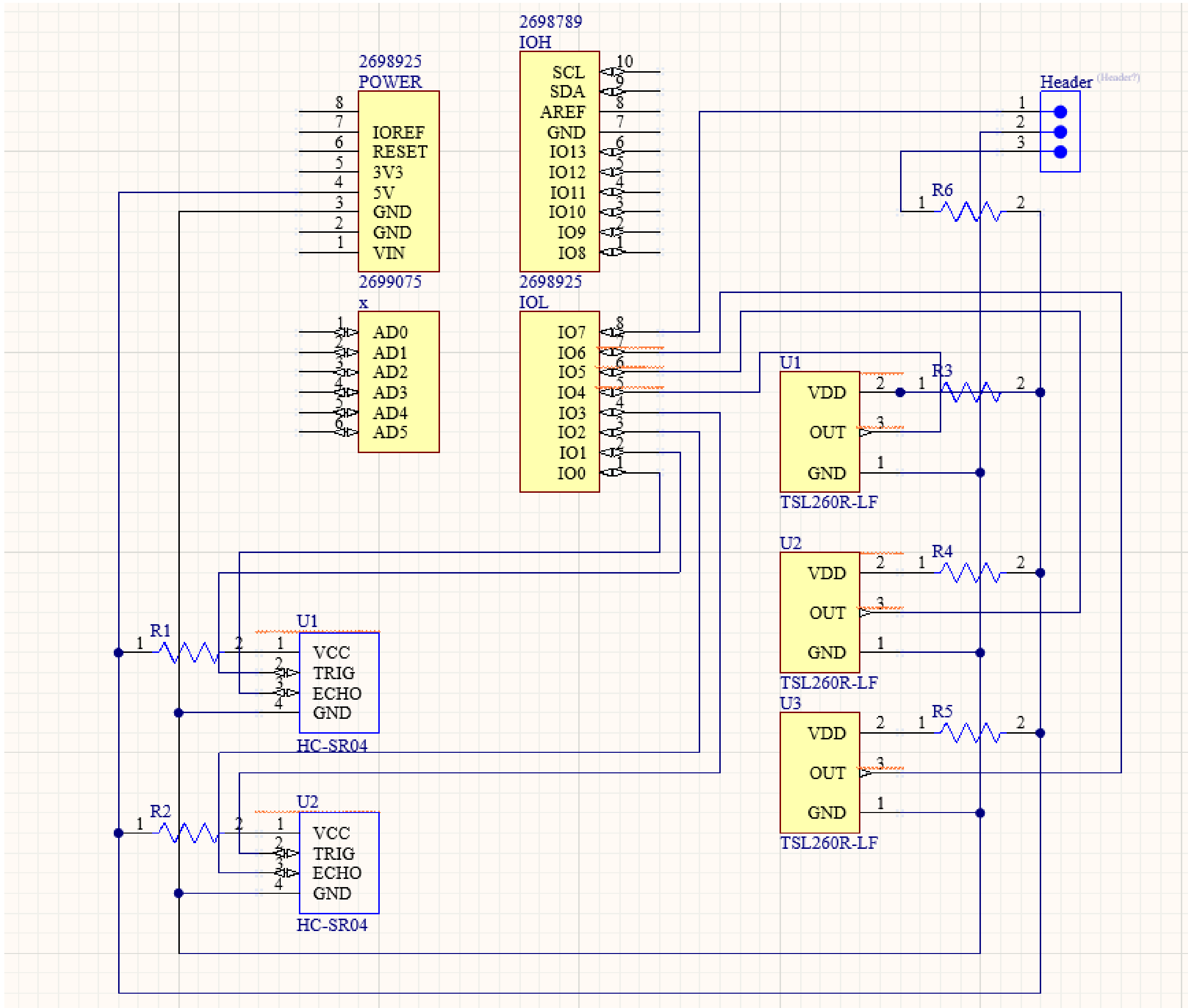


Figure 6.2 Altium Design

As seen above (figure number), we have our schematic of how the sensors will be connected to the Arduino. The sensors on the right are the ultrasonic sensors which are connected by resistors from the 5V and then connected to the pins on the Arduino. The sensors on the right are the IR (InfraRed) sensors that are connected to resistors and require 3 pins total on the Arduino for the data. The 3 Header female pins are for the tilt sensor.

6.3 Electronic

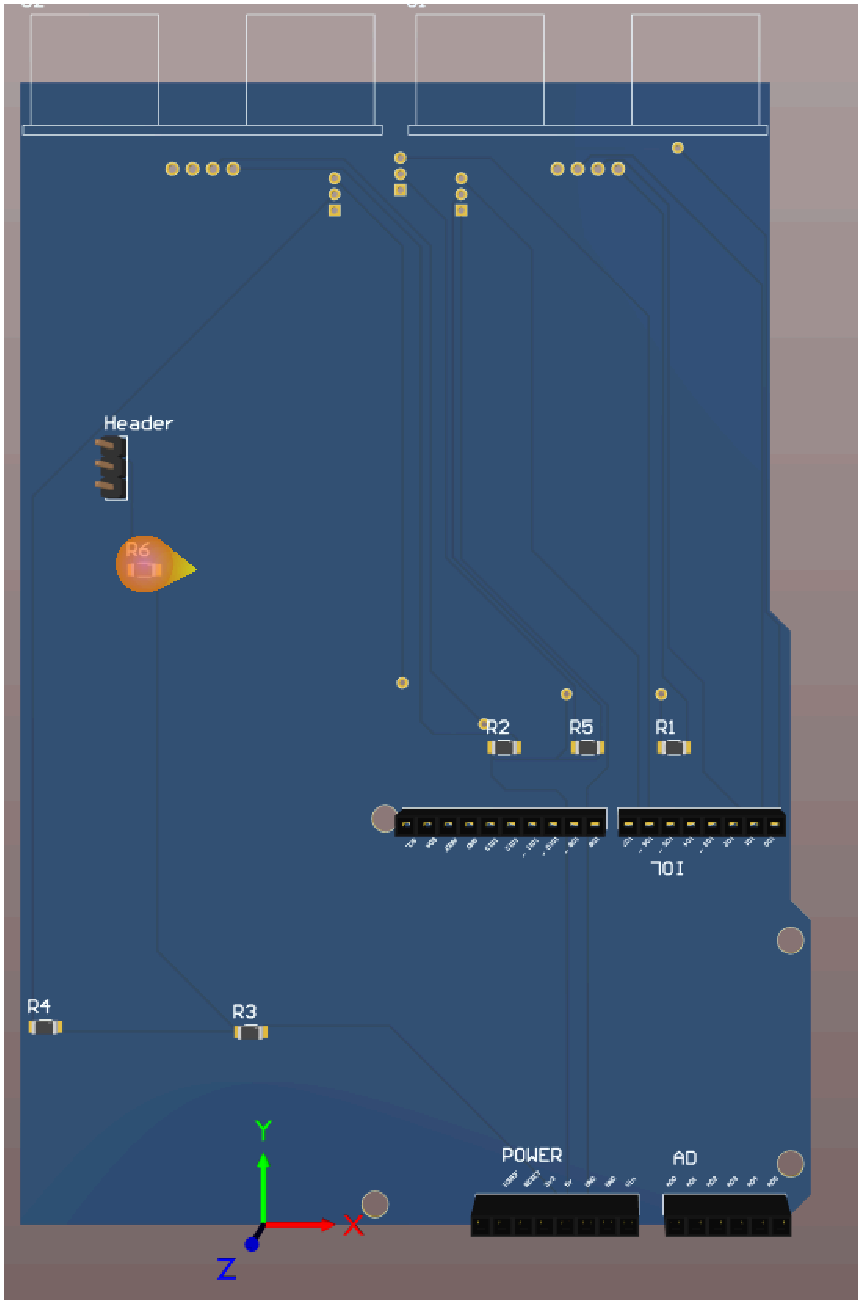


Figure 6.3 PCB Design

The 3D model above (figure number) is the 3D view of how the sensors will be placed on the board. The front sensors are the ultrasonic sensors, they are in front of the vehicle to detect any obstacle in front of it. We then have the 3 IR sensors underneath the board because they would need to detect the white line. In top left, we have the 3 female header pins for the tilt sensor, this will help to detect whether there is an incline when the car is advancing.

6.4 Software

The pseudocode below describes how the car and its input-detecting sensors can perform actions on what the sensor reads. If the IR sensor no longer detects a line, it will perform a stop and evaluate if it is a permanent or temporary stop.

```
C/C++
 read_ir_sensor();


start infinite loop:
if read_ir_sensor() = #FFFFFF:
   continue on path
else:
   stop
   detect if stop is temporary (obstacle) or permanent (end).
   if end is detected:
      PERMA-STOP;
      exit loop;
```

The pseudo-code below illustrates the process of using the Ultrasound sensor to identify the obstacle and safely avoid it. It will first use the sensor to check if there is an obstacle ahead of the vehicle, if there is, the vehicle will perform an evasive move through its axle to avoid the obstacle and follow up with the track

```
C/C++
int read_Ultrasound_Sensor();


start inf loop
if read_Ultrasound_Sensor() > obstacle_threshold;
  car.perform_evasive_manoeuvre();
else:
  car move straight() or car.follow_line
end loop
```

The pseudo-code below illustrates the process of overcoming the aforementioned ramp. After the ramp is detected, the motor will increase its power to overcome the ramp and will return to the normal motor power once it got down of the ramp

```
C/C++
int read_tilt_Sensor()

start loop
if read_tilt_Sensor < ramp.threshold{
  motor.increasePower();
else
  car go;
```

References

Number	Type of References	Reference IEEE style
[1]	Web Article	[1] R. Hernandez, “Arduino Clock Speed: A Guide to different board speeds,” EmbedWiz.com: Discover the World of Embedded Systems with EmbedWiz, https://embedwiz.com/arduino-clock-speed/#pros-and-cons-of-fast-clock-speeds (accessed Mar. 25, 2024).
[2]	Datasheet	[2] Raspberry PI, “DATASHEET raspberry pi 4 model B,” Raspberry Pi 4 Datasheet, https://datasheets.raspberrypi.com/rpi4/raspberry-pi-4-datasheet.pdf (accessed Mar. 25, 2024).
[3]	Datasheet	[3] “A000066-datasheet.pdf,” Arduino UNO R3 Datasheet, https://docs.arduino.cc/resources/datasheets/A000066-datasheet.pdf (accessed Mar. 25, 2024).
[4]	Website	[4] A. ArduinoModules, “KY-020 Tilt Switch Module,” ArduinoModulesInfo, https://arduinomodules.info/ky-020-tilt-switch-module/ (accessed Mar. 25, 2024).
[5]	Online Website/Store	[5] Keyes, “Online shopping for popular electronics, fashion, Home & Garden, toys & sports, automobiles and more.,” AliExpress, https://best.aliexpress.com/ (accessed Mar. 25, 2024).
[6]	Datasheet	[6] ElecFreaks, “Ultrasonic ranging module HC - SR04,” Ultrasonic Ranging Module HC - SR04 , https://cdn.sparkfun.com/datasheets/Sensors/Proximity/HCSR04.pdf (accessed Mar. 25, 2024).

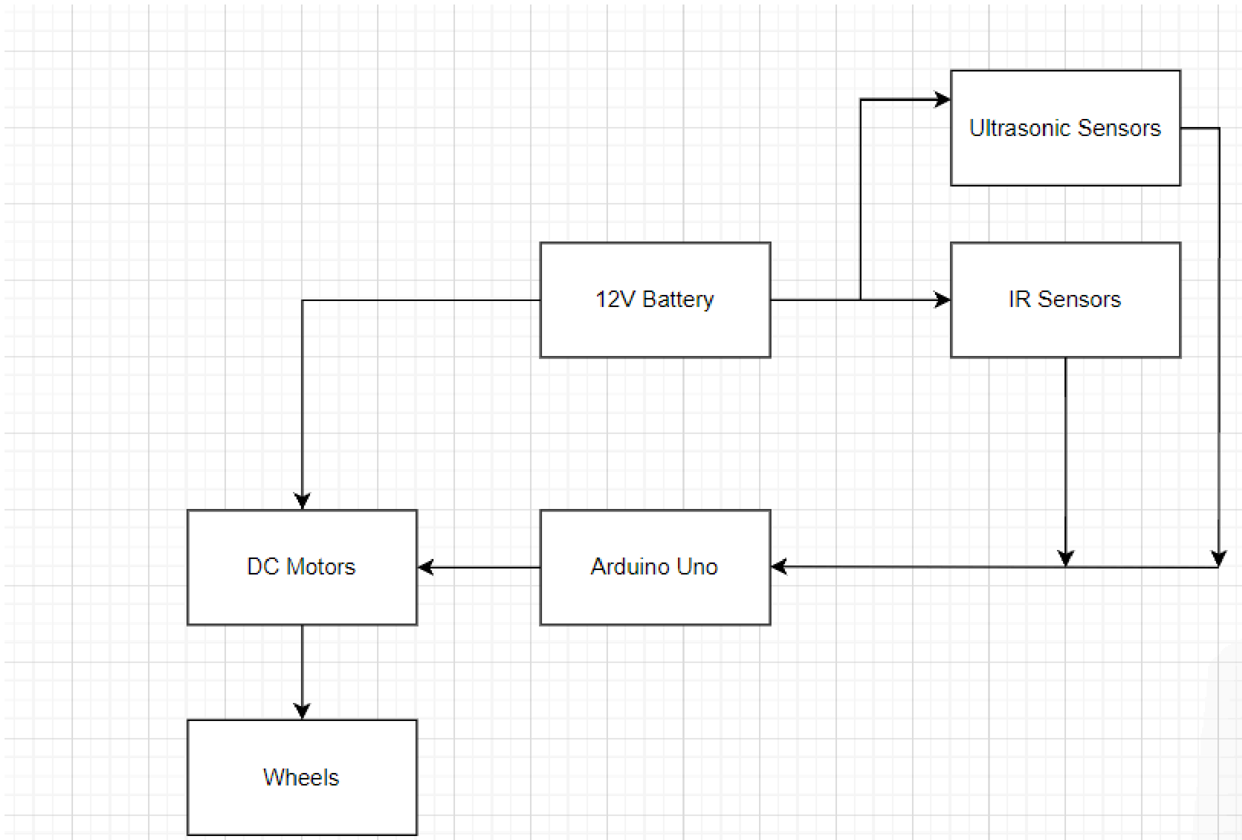
[7]	Website	[7] F. Nawazi, “HW201 infrared (IR) sensor module,” Circuits DIY, https://www.circuits-diy.com/hw201-infrared-ir-sensor-module/ (accessed Mar. 25, 2024).
[8]	Website	[8] I. I. S. Ibtinc, “What’s the difference between brushed and brushless motors?,” IBT Industrial Solutions, https://ibtinc.com/difference-between-brushed-brushless-motors/#:~:text=Brushed%20motors%20can%20operate%20at,on%20the%20same%20 battery%20 charge. (accessed Mar. 25, 2024).
[9]	Website	[9] G. Messick, “Inside NiMH: Understanding Nickel-Metal Hydride Batteries,” citizenside.com, https://citizenside.com/technology/inside-nimh-understanding-nickel-metal-hydride-batteries/ (accessed Mar. 25, 2024).
[10]	Website	[10] A. Szczepaniak, “Raspberry pi or Arduino – when to choose which?,” Leo Rover Blog, https://www.leorover.tech/post/raspberry-pi-or-arduino-when-to-choose-which (accessed Mar. 25, 2024).
[11]	Website	[11] I. Buckley, “Arduino Nano Pros and Cons: Is the cheapest Arduino worth it?,” MUO, https://www.makeuseof.com/tag/cheapest-arduino-nano/ (accessed Mar. 25, 2024).
[12]	Research paper	[12] “Advantages of Gyroscope: Disadvantages of gyroscope,” Advantages of Gyroscope | disadvantages of Gyroscope, https://www.rfwireless-world.com/Terminology/Advantages-and-Disadvantages-of-Gyroscope.html (accessed Mar. 25, 2024).
[13]	Website	[13] ArduinoModules, “KY-020 Tilt Switch Module,” ArduinoModulesInfo,

		https://arduinomodules.info/ky-020-tilt-switch-module/ (accessed Mar. 25, 2024).
[14]	Website	[14] ArduinoModules and Detlef, “KY-017 Mercury Tilt Switch Module,” ArduinoModulesInfo, https://arduinomodules.info/ky-017-mercury-switch-module/ (accessed Mar. 25, 2024).
[15]	Website	[15] E. Fahad, “KY-017 Mercury Tilt Sensor switch with Arduino Circuit Diagram & Code,” Electronic Clinic, https://www.electroniclinic.com/ky-017-mercury-tilt-sensor-switch-with-arduino-circuit-diagram-code/ (accessed Mar. 25, 2024).
[16]	Website	[16] D. Admin, “Obstacle detection sensor: Everything you need to know,” DreamVu, https://dreamvu.com/all-you-need-to-know-about-obstacle-detection-sensor/ (accessed Mar. 25, 2024).
[17]	Blog	[17] MaxBotix, “Ultrasonic sensors: Advantages and limitations,” MaxBotix, https://maxbotix.com/blogs/blog/advantages-limitations-ultrasonic-sensors (accessed Mar. 25, 2024).
[18]	Website	[18] F. FlyGuys, “Advantages and disadvantages of LIDAR Technology,” FlyGuys, https://flyguys.com/advantages-disadvantages-lidar-technology/ (accessed Mar. 25, 2024).
[19]	Website	[19] GfG, “Advantages and disadvantages of infrared sensor,” GeeksforGeeks, https://www.geeksforgeeks.org/advantages-and-disadvantages-of-infrared-sensor/ (accessed Mar. 25, 2024).

[20]	Article	[20] TRAX, “Wheels vs tracks: Advantages and disadvantages,” Lite Trax, https://litetrax.com/wheels-vs-tracks-advantages-disadvantages/ (accessed Mar. 25, 2024).
[21]	Article	[21] R. Schneider, “Electric motors: Brushed vs. brushless,” Galaxus, https://www.galaxus.ch/en/page/electric-motors-brushed-vs-brushless-12220 (accessed Mar. 25, 2024).
[22]	Article	[22] Ibtinc, “What’s the difference between brushed and brushless motors?,” IBT Industrial Solutions, https://ibtinc.com/difference-between-brushed-brushless-motors/ (accessed Mar. 25, 2024).
[23]	Article	[23] “What are the advantages and disadvantages of NiCad batteries?,” Computer Hope, https://www.computerhope.com/issues/ch000351.htm (accessed Mar. 25, 2024).
[24]	Blog	[24] “7 advantages that you should know for lithium ion batteries,” Energy Storage System Solutions Provider「POWEROAD」, https://www.poweroad.com/knowledges/7-advantages-that-you-should-know-for-lithium-ion-batteries/ (accessed Mar. 25, 2024).
[25]	Article	[25] “Pros and cons of alkaline batteries,” PROS AND CONS OF ALKALINE BATTERIES, [Online Components] https://www.onlinecomponents.com/en/blogpost/pros-and-cons-of-alkaline-batteries-224/ (accessed Mar. 25, 2024).
[26]	Website	[26] Conrad Electronic, “Conrad Energy nimh accupack 7.2 V 2000 Mah Aantal Cellen: 6 stick T-stekkersysteem kopen ? Conrad Electronic,” Conrad Electronic, https://www.conrad.nl/nl/p/conrad-energy-nimh-accupack-7-2-v-2000-mah-aantal-cellen-6-stick-t-stekkersysteem-2269467.html (accessed Mar. 25, 2024).
[27]	Website	[27] Absima, “Absima thrust eco 15 turn standard 540 size brushed motor - 2310060,” Wireless Madness,

		https://www.wirelessmadness.com/absima-thrust-eco-15-turn-standard-540-size-brushed-motor-2310060 (accessed Mar. 25, 2024).
[28]	Website	[28] AZDelivery, “Azdelivery 5 x IR infrared module for obstacle detection distance sensor compatible with Arduino and Raspberry Pi including e-book!,” Amazon.nl: Health & Personal Care, https://www.amazon.nl/-/en/dp/B07V9XD2R6/ref=twister_B0844YS4RL?_encoding=UTF8&th=1 (accessed Mar. 25, 2024).

Appendix A:



Ultrasonic Sensors
12V Battery
IR Sensors
DC Motors
Arduino Uno
Wheels