Czech Technical University

Theoretical Fundamentals of Mechanical Engineering

Department of Instrumentation and Control Engineering

FACULTY
OF MECHANICAL
ENGINEERING
CTU IN PRAGUE

Selection of Sensors and Actuators for a Small Autonomous Car Model

Bachelor Thesis

Sami Jradi

Supervisor: doc. Ing. Martin Novák, Ph.D.

Declaration of Authorship

I, Sami Jradi, declare that this thesis titled, 'Selection of Sensors and Actuators for a Small Autonomous Car Model' and the work presented in it are my own. I confirm that:

- This work was done wholly or mainly while in candidature for a bachelor degree at this University.
- Where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated.
- Where I have consulted the published work of others, this is always clearly attributed.
- Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work.
- I have acknowledged all main sources of help.
- Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself.

Signed:

Date:

“Imagination is more important than knowledge. Knowledge is limited. Imagination encircles the world.”

Albert Einstein

BACHELOR'S THESIS ASSIGNMENT

I. Personal and study details

Student's name: **Jradi Sami** Personal ID number: **467453**

Faculty / Institute: **Faculty of Mechanical Engineering**

Department / Institute: **Department of Instrumentation and Control Engineering**

Study program: **Theoretical Fundamentals of Mechanical Engineering**

Branch of study: **No Special Fields od Study**

II. Bachelor's thesis details

Bachelor's thesis title in English:

Selection of sensors and actuators for a small autonomous car model

Bachelor's thesis title in Czech:

Návrh senzorů a akčních členů zmenšeného modelu autonomního vozidla

Guidelines:

1) Search for sensors and actuators suitable for a small autonomous car model
2) Based on the search, select suitable sensors
3) Install the sensors on the car model, connect them and test
4) Program a simple algorithm for obstacle detection with a reaction from the car

Bibliography / sources:

[1] Thor I. Fossen, Sensing and Control for Autonomous Vehicles: Applications to Land, Water and Air Vehicles (Lecture Notes in Control and Information Sciences) 1st ed. 2017 Edition, Springer, ISBN-13: 978-3319553719

[2] Bill Schweber, The Autonomous Car: A Diverse Array of Sensors Drives Navigation, Driving, and Performance <online>, available <https://cz.mouser.com/applications/autonomous-car-sensors-drive-performance/>, accesed 8.3.2019

Name and workplace of bachelor's thesis supervisor:

doc. Ing. Martin Novák, Ph.D., Division of electrotechnics, FME

Name and workplace of second bachelor's thesis supervisor or consultant:

Date of bachelor's thesis assignment: **26.04.2019** Deadline for bachelor thesis submission: **12.06.2019**

Assignment valid until: ____________

doc. Ing. Martin Novák, Ph.D.
Supervisor's signature

Head of department's signature

prof. Ing. Michael Valášek, DrSc.
Dean's signature

III. Assignment receipt

The student acknowledges that the bachelor's thesis is an individual work. The student must produce his thesis without the assistance of others, with the exception of provided consultations. Within the bachelor's thesis, the author must state the names of consultants and include a list of references.

26.04.2019
Date of assignment receipt

Student's signature

Czech Technical University

Abstract

Faculty of Mechanical Engineering
Department of Instrumentation and Control Engineering

Bachelor of Science

By Sami Jradi

This thesis is based on the research of the technology of sensors and the experimentation with sensors and actuators for building a model of an autonomous car. Sensors are selected and mounted on a small car model by holders which are designed and 3D printed. They are then connected to a microcontroller board. The next step is programming the board to read inputs from the sensors and send signals to trigger the actuators depending on the condition and algorithm defined. As a result, the autonomous system needs to sense the environment, determine the exact position on the road, and decide how it should behave in a given situation. Thus, combining sensor physics and the mechanical actuation of the vehicle by a software. The last part focuses on testing the built model inside a maze and documenting the results in terms of charts showing the detection done by the sensors and their effect on the movement of the car. In the end a conclusion is made based on the work done in the project and the results obtained, in addition to an overview of future improvements and possibilities of producing a fully autonomous vehicle for consumers.

Acknowledgements

I would like to thank my supervisor doc. Ing. Martin Novák, Ph.D. with the help, knowledge and guidance provided for me to be able to develop this thesis. At the same time, I would like to thank the Czech Technical University in Prague for providing me with the required tools and equipment for completing this project, as well as giving me the opportunity to complete my bachelor's degree in "Theoretical Fundamentals of Mechanical Engineering".

Also, I'd like to thank the professors of the university that have helped me throughout my studies and provided me with the required knowledge to complete my thesis. This project is a reflection of all this knowledge I have been given during my period in Prague.

I would like to thank my family and friends that have supported me during my time in Prague, which has also contributed greatly in producing this work.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Autonomous cars, or in other words self-driving cars, have been a dream for a while. In fact, since as far as the 1920s several experiments have taken place regarding this topic. The first signs throughout history concerning the development of self-driving vehicles was the radio-controlled "American Wonder". The vehicle was in fact a Chandler Motor car which was equipped with a transmitting antennae and was operated by a second car that followed it and sent out radio impulses which were caught by the transmitting antennae. The antennae would then forward the signals to circuit-breakers which operated small electric motors that directed every movement of the car. [1]

However, the real breakthrough dates back to Futurama, an exhibit at the 1939 New York World's Fair, when General Motors demonstrated its vision of a futuristic world which included automated highway systems that would guide self-driving cars. Now a days, cars do depend on several autonomous features, but the dream of building full-fledged autonomous vehicles is still alive to this day. [2]

The content of this thesis will be focusing heavily on understanding the function of different types of sensors and identifying the suitability of these sensors for building a small autonomous car model.

1.1 Sensors

Sensors are sophisticated devices that detect and respond to some type of input from the physical environment and convert it into a signal which can be measured electrically. The specific input could be light, heat, motion, moisture, pressure, or any one of a great number of other environmental phenomena. There are certain features which have to be considered when choosing a sensor such as accuracy, environmental condition - usually has limits for temperature/humidity, range - measurement limit of sensor, calibration - essential for most of the measuring devices as the readings changes with time, resolution - smallest increment detected by the sensor, cost and repeatability – the reading that varies is repeatedly measured under the same environment. For the car model, four different types of sensors have been chosen to be examined and compared in order to make a final selection of what is required for this project.

1.1.1 Lidar Sensors

Since originating in the 1970s, Lidar technology has been widely implemented in several sectors. Its first application came in meteorology, where the National Center for Atmospheric Research used it to measure clouds. However, the accuracy and usefulness of Lidar systems became publicly realized in 1971 during the Apollo 15 mission, when astronauts used a laser altimeter to map the surface of the moon. [3]
Lidar, which stands for Light Detection and Ranging, measures the distance by firing rapid pulses of laser light, usually up to 150,000 pulses of either visible ultraviolet or near infrared light, at a target. When the light hits the target, it gets reflected back to the sensor which then measures the time taken for the pulse to bounce back from the target. The distance is then deduced by using the speed of light to calculate the distance traveled accurately using the relation,

$$d = \frac{c\ t}{2} \qquad (1)$$

Where "d" is the distance to the target, "c" is the speed light and "t" is the time of flight of the laser light.
The result is precise three-dimensional information about the target object and its surface characteristics. In the case of self-driving cars, Lidar is used to generate huge 3D maps that the car can then navigate through. High end Lidars can even identify the details of a few centimeters at a distance of more than 100 meters. For example, not only can it detect pedestrians but it can also tell which direction they're facing. Of course, such autonomous technology isn't without its downfalls – take for example the fatal accident by an Uber self-driving car in Arizona where the technology failed to pick up a pedestrian crossing the road. [4]
However, as Lidar becomes more sophisticated, it will be increasingly capable of detecting and tracking objects. Improvements will mean higher resolution imagery will be possible and it will be able to operate at longer ranges so that the technology is capable of differentiating between someone walking, and someone on a bike, their speed, and direction.

Choosing a suitable Lidar sensor for the project requires deep research and analysis of the suitability of different models of Lidar sensors, in addition to availability in the Czech Republic. One of the main specifications is the "working Range" of the sensor, as in the minimum and maximum distance the laser beam can travel to the target and be reflected back while keeping in mind that the longer a beam travels, the more it will alternate as it gets reflected by the target. Another very important sensor property is of course the "Distance Resolution" which can tell us the smallest change in distance that the sensor can detect. The "Scanning Frequency" of a Lidar sensor tells us the period of time required for one full scan of a complete line when the target is being scanned sequentially using a laser beam. On the other hand, in signal processing, sampling is a very crucial mathematical operation which performs the reduction of continuous-time signals to discrete-time signals. “Sampling Rate” is the average number of samples, a set of values at a point in time or space, which are processed in one second. The final criteria is the price which has been converted to Czech Koruna.

Model	Working Range [m]	Distance Resolution [mm]	Scanning Frequency [Hz]	Sampling Rate [times/s]	Price [CZK]
YDLIDAR X4 360° Laser Scanner [5]	0.15 ~ 11	< 0.5	6 ~ 12	5000	2300
YDLIDAR G4 360° Laser Scanner [6]	0.015 ~ 16	< 0.5	5 ~ 12	9000	7460
RPLDIAR A2M8 360 Degree Laser Scanner [7]	0.15 ~ 12	< 0.5	5 ~ 15	8000	6830

Table 1 Lidar sensor models

1.1.2 Radar Sensors

Radar has been around for as long as 1904 when German inventor Christian Huelsmeyer invented the “telemobiloscope”, which was an early implementation of the radar technology that could detect ships up to 3000 m away. Initially it could only detect the presence of an object, but sooner rather than later newer were able to determine the distance as well. Almost one hundred years into the future, Radar in now able to measure velocities, angles and distances of objects in land, sea and in air. [8]

Radio Detection and Ranging, or in other words Radar, transmits short radio pulses with very high pulse power focused in one direction by the directivity of an antenna and propagates at the speed of light. These pulse signals, when moving in the direction of an object, will scatter in all directions expect a very small portion that gets reflected back. The antenna doubles up as a receiver as well as a transmitter using an important piece of equipment in the radar apparatus called a “duplexer”. The duplexer cause the antenna to swap back and forth between being a transmitter and a receiver. While the antenna is transmitting, it cannot receive and vice-versa. The Radar then evaluates the information received and determines the distance to the object knowing that the propagation of radio waves happens at the speed of light. In other words, the Radar works similarly as Lidar, except it uses radio waves instead of light waves and measures the time taken for the radio waves to return to the antenna. Then the same equation (1) can be applied to determine the distance.

Although similar to light waves, radio waves are much longer and have much lower frequencies. Light waves have wavelengths of about 450 ~ 750 nanometers, whereas the radio waves used by Radar typically range from about few centimeter to a meter, roughly a million times longer than light waves. Another advantage of the Radar is that they're particularly useful in bad weather conditions, capable of working in fog, snow, rain and darkness that otherwise would compromise Lidar sensors.

We have been able to develop Radar technology to a point where it's playing a key role in the driving of vehicles in the form of Advanced Driver Assistance Systems (ADAS), which is a system that is developed to automate, adapt and enhance vehicle systems for safety and better driving. Thus, reducing accidents caused by human errors. This system constitutes an intermediate stage in the development of autonomous vehicles. [9]

A special class of the Radar technology is the millimeter-wave or mmWave Radars. These Radar sensors transmit signals with a wavelength in the millimeter range which is considered an extremely short wavelength in the electromagnetic spectrum. One of their biggest advantages is the ability to detect movements as small as fractions of a millimeter in addition to having a high accuracy. Texas Instruments have been able to develop this technology to the level of implementing the Frequency-Modulated Continuous wave (FMCW). The FMCW Radar sends a frequency-modulated signal continuously which differs from the traditional periodic pulsed signals. The mmWave Radars from Texas Instruments are usually in the form of integrated single-chip FMCW transceiver or as an evaluation board equipped with the sensor chip. The mmWave sensors are classified based on their number of receivers, number of transmitters, max sampling rate, Intermediate frequency (IF) bandwidth. [10]

Model	**Number of Receivers**	**Number of transmitters**	**Max Sampling Rate [Msps]**	**IF bandwidth [MHz]**	**Price [CZK]**
TI AWR1243BOOST [11]	4	3	37.5	15	7756
TI AWR1443BOOST [12]	4	3	12.5	5	7756
TI AWR1642BOOST [13]	4	2	12.5	5	7756

Table 2 mmWave sensor models

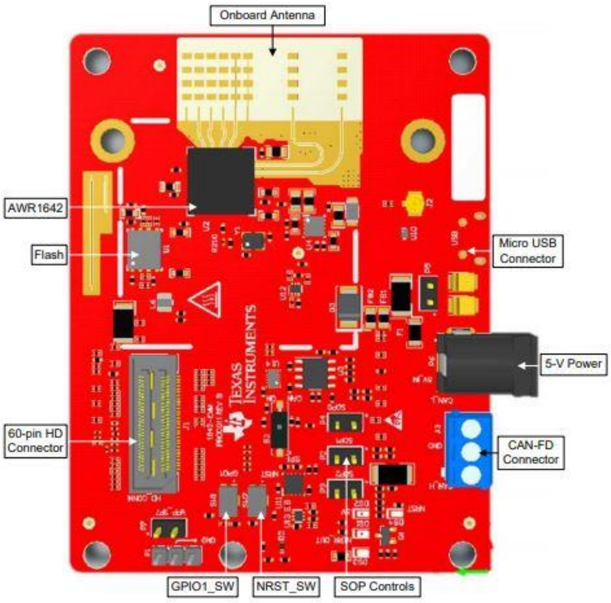

Figure 1 An example of the AWR1642BOOST Development Kit for FMCW Radars [10]

1.1.3 Ultrasonic Sensors

The history of the ultrasonic sensors dates back to 1826, when Swiss physicist Jean-Daniel Colladon discovered sonography using an underwater bell and accurately determined the speed of sound in water. The next scientific breakthrough didn't come until sixty years later when famous French physicist Pierre Curie started experimenting with the relationship between electrical voltage and pressure on crystalline material, leading to the "The Piezoelectric Effect" and setting the stage for modern ultrasonic transducers. The first ultrasonic transducer was the hydrophone. Inspired by the Titanic tragedy, it was invented to detect icebergs and help navigate ships by sending and receiving low frequency sound waves.

Ultrasound waves are similar to sound waves, where both travel through a medium. Ultrasound waves consist of high-frequency sound waves that are inaudible to human beings. The frequency of the ultrasound waves is normally above 20 kHz. However, some creatures such as bats can hear as well as generate the high frequency ultrasound waves. In fact, bats are blind from the eyes yet, thanks to ultrasonic ranging, they have a vision so precise that could distinguish between the smallest objects in nature, whether it's an ant or a leaf.

An Ultrasonic sensor is an instrument that uses a transducer to send ultrasonic pulses in order to measure the distance to an object. As the sound waves travel through the air and eventually making contact with the object. These ultrasonic pulses are then reflected off the object and the echo is received by the transducer. The distance to the object is determined by measuring the time lapses between the sending and receiving of the ultrasonic pulse using the relation

$$d = \frac{v\,t}{2} \tag{2}$$

Where, “t” is the time taken by the waves to reach back to the sensor, “v” is the speed of sound in the medium. [14]

Ultrasonic sensors have become the basis for parking and maneuvering systems in modern vehicles. As parking sensors, they use the same basis as the sonar by detecting obstacles when parking but nowadays the technology is evolving into an automatic parking system. The system supports emergency braking functions at low speeds when detecting very close objects, but in the case suddenly emerging obstacles such as pedestrians, the response is much faster. [15]

When choosing a basic Ultrasonic sensor, there are certain specifications to be inspected. The “Working range” is the first parameter that is usually checked in order to define whether the sensor can detect objects within the range of the experimental application. The “Distance Resolution” then becomes an important factor for detecting the smallest change in distance that the sensor can pick up. Since Ultrasonic sensors do not just detect objects directly ahead but also objects offset at some angle, then the “Effectual Angle” is needed to specify the range of angles over which detection occurs.

Model	Working Range [m]	Distance Resolution [mm]	Effectual Angle [°]	Price [CZK]
Devantech SRF05 [16]	0.01 ~ 4	30	<15 °	400
HC-SR04 [17]	0.02 ~ 5	10	<15 °	80
DFRobot Weatherproof [18]	0.25 ~ 4.5	5	< 70°	440

Table 3 Ultrasonic sensor models

1.1.4 Optical Flow Sensors

The theory of optical flow first originated from American psychologist James J. Gibson in the 1940s. His aim was to describe the visual stimulus for the perception of movement by the observer in the world. This so called perception of movement revolves around the perception of the shape, distance and movement of objects in the world, as well as the control of motion. Gibson emphasized on the ability of the observer to recognize possibilities for action within the environment. [19]

The optical flow later developed as a method to estimate motion from a sequence of ordered images as either instantaneous image velocities or discrete image displacements. However, optical flow does not directly measure the distance to an object. Additional methods are applied in order to estimate the distance. One method is done by adding sensors that directly or indirectly measure the distance. Another method is a differential method which uses local Taylor series approximations of the image signal and the partial derivatives with respect to the spatial and temporal coordinates. [20]

The Optical flow sensor technology, based on the optical flow theory by Gibson, is capable of measuring visual motion and outputting a measurement based on optical flow. Optical flow sensors exist in various setups. An image sensor chip connected to a processor programmed to run an optical flow algorithm is a typical setup. Another setup relies on a vision chip, which is an integrated circuit having both the image sensor and the processor on the same die. [21]

Optical flow sensors have been instrumental in stability and obstacle avoidance in unmanned aerial vehicles, commonly known as a drone. In addition, Optical flow sensors are typically used in an optical computer mouse, where a processing circuitry is implemented using either analog or mixed signal circuits to enable fast optical flow computation using minimal current consumption. [22]

As a matter of fact, the standard optical mouse contains all of the hardware necessary for tracking X/Y movement on a flat surface. Due to its simple interface, the Optical Flow sensor in a mouse can be easily manipulated. Usually it has a low pixel frame but a high frame update rate for generating smooth flow of the mouse cursor. The type of lens and field of view are also important factors considered when it comes to choosing a suitable Optical sensor for distance measurement and obstacle detection.

Model	**Pixel Frame**	**Update Rate [fps]**	**Lens**	**Field of View [°]**	**Price [CZK]**
ADNS-3080 [23]	30 x 30	2000 ~ 6400	M12x0.5	11	451
CUAV PX4FLOW 2.1 [24]	640x480	30	M12x0.5	115°	1460
Arducam OV5647 [25]	2592 x 1944	30	M12	10°	270

Table 4 Optical flow sensor models

1.2 Summary

The main focus in the first chapter was on four different kinds of sensors which can be implemented in an autonomous vehicle and detect the distance between the vehicle and other obstacles. These sensors, while they serve the required purpose in terms of functionality, differ slightly in terms of installation, specifications and programmability. The most powerful and dynamic sensors from the selected models seem to be the mmWave Radar sensors from Texas Instruments. However, when taking into consideration their specifications and price range, they seem a bit excessive and thus unsuitable for our basic autonomous car model. On the other hand, Optical flow sensors are very efficient and widely applicable nowadays in many fields where there is a need to measure the visual motion or relative motion between a vehicle and other objects in its vicinity. As for the suitability of sensors to be used in a small autonomous car model, a combination of both Lidar and Ultrasonic sensors were chosen due to their easy applicability and direct functionality. The aim of such an experiment is the connection and installation of sensors to the car model by means of an Arduino microcontroller. The Arduino microcontroller's main function is to read the inputs from the sensors and produce an output in the form of a response. For the output response, it will be required to program the board using its custom Arduino programming language.

Chapter 2

2.1 Equipment and Components

This chapter will be focusing more on the technical side of the project where several components that are required for the car model will be discussed thoroughly. Components such as the ultrasonic sensor, Lidar sensor, microcontroller, electric motor and servo motor will be essential to the functionality of the car model.

2.1.1 Ultrasonic Sensor

Figure 2 HC-SR04 Ultrasonic Sensor [26]

For this model the HC-SR04 was chosen to serve as the main sensor to measuring the distance between the car and possible obstacles. The HC-SR04 consists of four connecting pins: VCC (5V power supply), Trig (trigger input pin), Echo (receiver output pin) and GND (power ground). The distance measurement is initiated when the trigger pin receives a high pulse for at least 10 μs, the transmitter will then emit 8 bursts of ultrasonic waves at 40 kHz and traveling at velocity of 340 m/s. As the waves encounter an obstacle, they are then reflected back and received by the echo pin. [26]

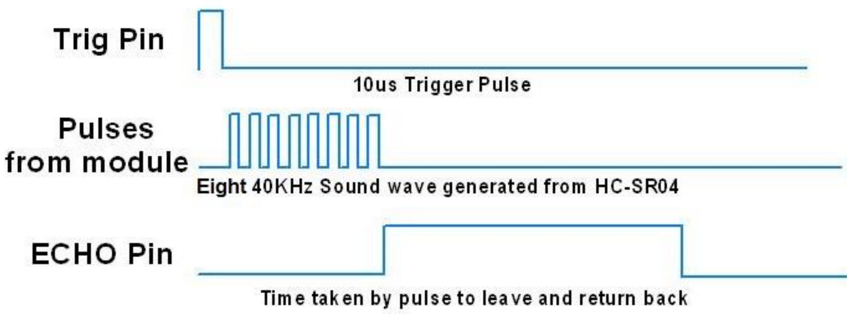

Figure 3 Ultrasonic HC-SR04 Timing Diagram [26]

Four HC-SR04 ultrasonic sensors will be mounted on front, left, right and rear sides of the car to ensure full coverage of the surroundings. For good fixation and stability of these sensors, it was necessary to design custom holders using CAD software. Four copies of the holders were produced using a 3D printer.

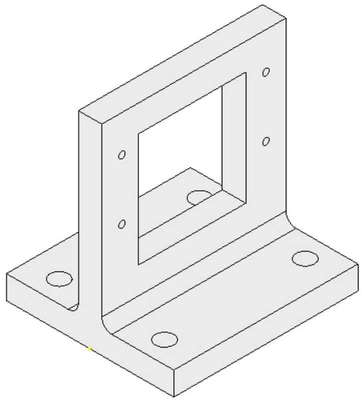

Figure 4 3D Model of Ultrasonic sensor Holder

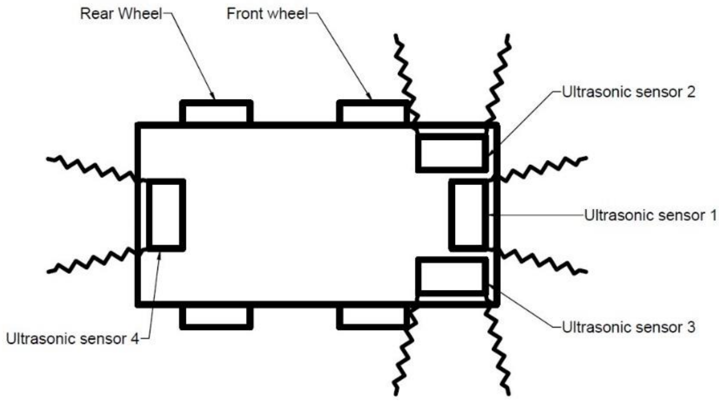

Figure 5 Sketch of how the sensors are supposed to mounted on the model (Measuring angle is 15°)

2.1.2 Lidar Sensor

Figure 6 The HLS-LFCD2 360 Degree Laser Scanner [27]

The other sensor to be used in this project is the HLS-LFCD2 which consists of the distance sensor attached on a rotating base. This sensor is a line sensor capable of sensing in 360 degrees and collecting a set of data to be used for navigation. For each successful measurement, the measured distance is sent with the according angle. Much similar to the ultrasonic sensor, this Lidar has a VCC and GND pins in addition to the TXD pin (transmit data), the RXD pin (receive data), the MOTOCTL pin (motor control input) and the VMOTO pin (5V DC motor power supply). Since this sensor is mounted on a rotating base and capable of measuring in a 360 degree angle, then one HLS-LFCD2 is enough to be used in the project. However, a holder was needed to be designed to properly mount the sensor on top of the car from the front side. The holder must provide good stability for the sensor as well as sufficient height to prevent any components of the car interfering with the readings. [27]

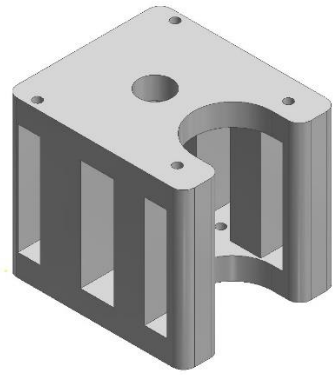

Figure 7 3D model of Lidar sensor holder

2.1.3 Microcontroller

A microcontroller is a single integrated circuit which can be seen as a small computer. A typical microcontroller consists of a processor, memory and input/output peripherals. It is designed to implement certain operations in an embedded system. A microcontroller will run just one program repeatedly and not a full operating system without requiring additional computing components, because they are designed with sufficient onboard memory as well as containing general I/O pins for operations which enables them to directly interface with sensors and other programmable components. It mainly gathers input, processes the information and outputs a certain action based on the information gathered. [28]

For the car model, an Arduino Mega 2560 single-board microcontroller is used. Arduino is an open-source hardware and software project that can be used to automate devices that interact with the world. The Mega 2560 board is used for complex projects such as robotics projects and comes with more memory space and I/O pins as compared to other microcontroller boards available in the market.

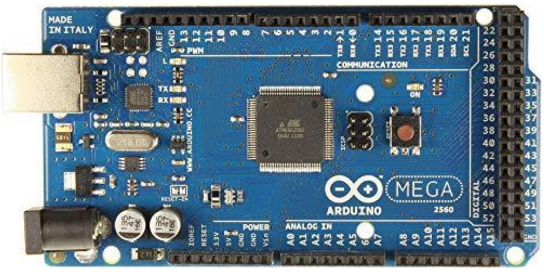

Figure 8 Arduino Mega 2560 Microcontroller Board [29]

For programming the board, it is preferable to use the official integrated development environment (IDE) for the Arduino which supports C programming. The code, which is also called a sketch, is then compiled in the software and transferred to the board by USB cable. The Arduino Mega 2560 comes preprogrammed with a bootloader that allows uploading of the new code without the use of an external hardware programmer. The bootloader is a section of the program memory which runs before the main code runs. First, it checks if the computer is trying to program it and uploads the program into the memory (without overwriting the bootloaders memory). However, if the computer is not trying to upload a code then it tells the microcontroller chip to run the code that's already stored in the memory. [29]

2.1.4 Electric Motor

An electric motor is basically a device which is used to convert electric energy into mechanical energy. The motor depends on magnetism to create motion. According to the fundamental law of magnets, opposite poles attract while similar poles repel each other. These attracting and repelling forces are what create the rotational motion of the motor. Two main components are the rotor, which is and electromagnet, and the field magnet, which is usually a permanent magnet in small motors.

The car model will be equipped with the SGM25F-370 motor. This type of motor is a DC geared motor which is a combination of a motor and a gearbox. The purpose of the geared motor is to increase torque while at the same time reducing the speed. This concept is known as gear reduction. [30]

Operating Voltage	9-14V (12V)
Nominal Current	300 mA
Nominal Speed	37 rpm
Nominal Torque	258 Nmm

Table 5 SGM25F-370 DC Geared Motor specifications [31]

Figure 9 SGM25F-370 DC Geared Motor [31]

For controlling the motor, an L298N H-bridge motor driver is required. The role of the H-bridge is to allow speed and direction control of the DC motor. It contains four switching elements and a load at the center in an H-like configuration. When two switches are closed a positive voltage will be applied across the motor. However, by opening these switches and closing the other two, the voltage is reversed to allow reverse operation of the motor.

Logical Voltage	5V
Drive Voltage	5V-35V
Logical Current	0-36 mA
Drive current	2 A
Max Power	25 W

Table 6 The L298N H-Bridge specifications [32]

Figure 10 The L298N H-bridge [32]

2.1.5 Servo Motor

A servo motor is a type of motor which allows the precise control of angular position. It consists of three basic parts which are the motor, the potentiometer and the control board. The principle of the servo motor depends on a control circuit which can precisely regulate how much movement there is and in which direction as the potentiometer's resistance changes. When the shaft of the motor reaches the desired position, power supplied to the motor is shut off. This position is sent by electrical pulses of variable width, also known as pulse width modulation (PWM), through the wires while the speed of the motor is proportional to the difference between its actual position and the target position. In other words, as the motor reaches closer to its desired position then the speed of rotation of the shaft becomes slower.

In this project, the Hitec HS-311 Servo Motor model was chosen for the precise control of the steering of the car's front wheels. The servo motor, when connected to the microcontroller along with the sensors, will be programmed to determine the direction in which the car would need to move.

Voltage Range	4.8V - 6.0V
No-Load Speed	0.19 sec/60°
Max PWM Signal Range	575-2460 μsec
Max Travel	202.5°

Table 7 Hitech HS-311 Servo Motor specifications [33]

Figure 11Hitec HS-311 Servo Motor [33]

2.2 Connections and Assembly

After discussing the main components needed in the previous subchapter, this thesis will now dive into the electrical assembly of the components and the programming of the microcontroller board.

2.2.1 Electrical Assembly of Ultrasonic Sensor

Since there are several components which are required to be connected to the Arduino Mega 2560, for this purpose an Arduino Mega sensor shield is preferred to be used. The sensor shield works as an extension to the Arduino Mega and is mounted on top. It provides a significantly higher number of additional pins which is what is required in this project in order to connect all of the components. The pins are divided into sets of threes (a signal pin, a voltage pin and a ground pin).

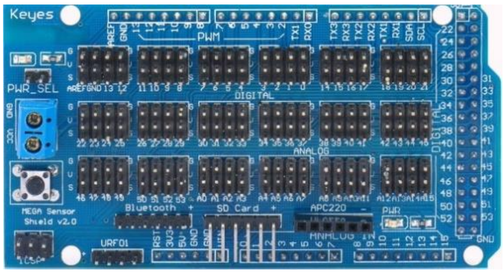

Figure 12 Arduino Mega sensor shield [34]

Using jump wires, the components will be connected to the shield which is mounted on the microcontroller board. The connections will be detailed in the following table:

Component	Location of Signal Pins
Ultrasonic sensor 1 (trigpin, echopin)	2, 3
Ultrasonic sensor 2 (trigpin, echopin)	18, 19
Ultrasonic sensor 3 (trigpin, echopin)	44, 45
Ultrasonic sensor 4 (trigpin, echopin)	8, 9
L298N H-bridge (EnA, In1, In2)	46. 47 .48
Hitec HS-311 Servo Motor	52

Table 8 Connection of the components to the specific pins on the shield (Ultrasonic Sensor)

As previously discussed in this thesis and according to what is shown in Table 5, each ultrasonic sensor will have a trigger (transmitter) pin and an echo (receiver) pin to be connected in addition to both the VCC (voltage common collector) and GND (ground) pins. While L298N H-bridge has three separate connections other than voltage and ground. The Enable A pin is used for enabling and controlling the speed of the motor which is to be connected to the terminals A of the H-bridge, while the purpose of both Input 1 and Input 2 logic pins is to determine the direction of rotation of the motor. The H-bridge will also serve the purpose of powering up the Arduino board by connecting a DC power supply to its voltage and ground terminals and then supplying the Arduino power with power through wires. Finally, the servo motor has three total connections only such as the yellow wire which has to be connected to a digital pin on the Arduino and the orange wire and black wire to both the voltage and ground pins respectively.

2.1.2 Electrical Assembly of Lidar Sensor

Much like in the case of the ultrasonic sensor, the Arduino Mega sensor shield is used for providing additional pins to the Arduino Mega 2560. The connections of the both the L298N H-bridge and the servo motor are unchanged while the connections of the Lidar sensor and the motor controlling the rotating base are detailed in the table below. **An important note is that for this Lidar model, a 5V to 3.3V level shifter is needed for both the TXD and RXD pins.**

Component	Location of Signal Pins
Lidar (TXD, RXD, MOTOCTL)	0, 1, 2
L298N H-bridge (EnA, In1, In2)	46. 47 .48
Hitec HS-311 Servo Motor	52

Table 9 Connection of the components to the specific pins on the shield (Lidar Sensor)

The TXD pin and RXD pin, or the transmitter pin and receiver pin, have to be connected to serial ports to 4 and 5 pin respectively which are predetermined on the Arduino board as the serial communication ports. These serial ports are known as Universal Asynchronous Receiver/Transmitter (UART) and are physical circuits in a microcontroller. The UART's main purpose is to transmit and receive serial data. The transmitting UART converts parallel data into serial form before transmitting it to the receiving UART before its converted back to parallel data. The reason these serial ports are called asynchronous is because the transmitting UART adds start and stop bits to the data being transferred instead of using a clock signal. The purpose of these bits is to define the beginning and the end of the transferred data which enables the receiving UART to know when to start reading the bits. As soon as a start bit is detected, it begins reading the incoming bits at a specific frequency known as the baud rate which will be set in the code. The baud rate is defined as the rate at which data is transferred and expressed in bits per second (bps). While the use of UARTs is very advantageous, they can be let down by some disadvantages such as having the size of the data frame limited to a maximum of 9 bits. [35]

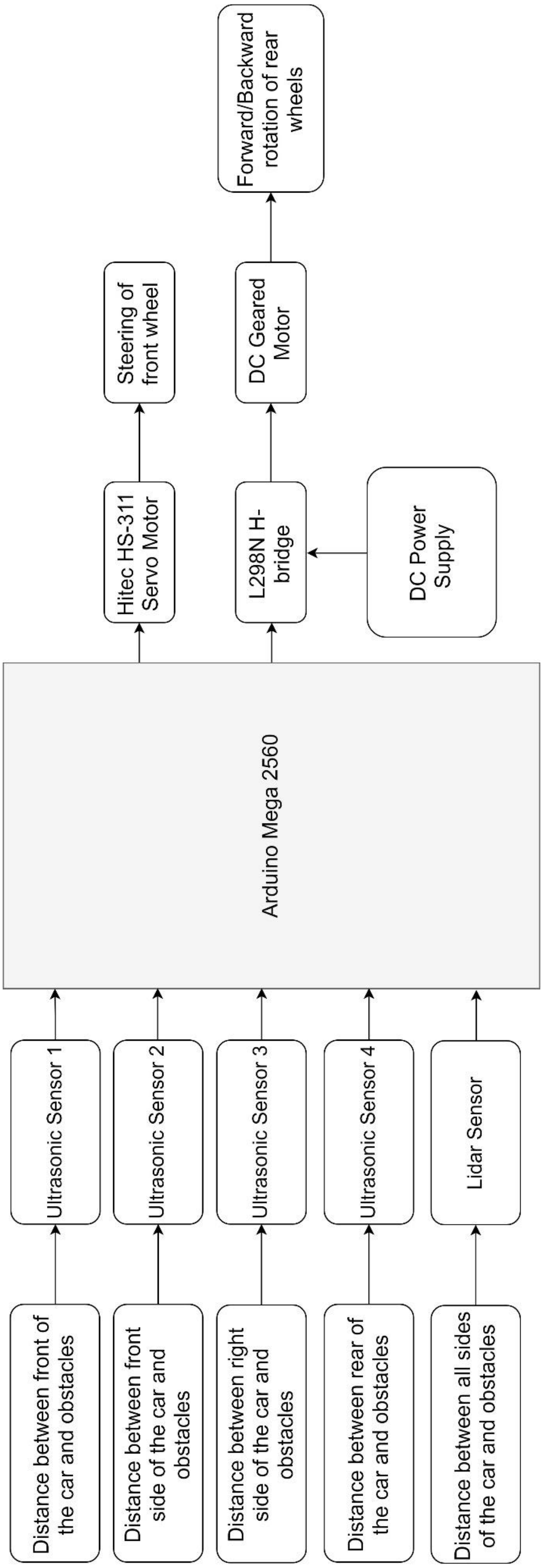

Figure 13 Block diagram showing all the connections and their respective purpose

2.2.2 Algorithms and Programming

After assembling all of the components and mounting them on top on the car model, the next important step will be programming of the microcontroller. For Arduino boards, it is recommended, but not mandatory, to use the Arduino software which is also known as the IDE (integrated development environment). The Arduino IDE provides the ability to write programs and upload them directly to the board to be executed. These programs are written using the Arduino language which is based on a set of C++ functions that are called upon from preinstalled libraries and passed directly to a C++ compiler. When programming the Arduino board, a very simple structure is used. The Arduino program consists mainly of two blocks, preparation and execution, which contain a set of statements each.

The preparation block is known as the setup function which is the first to be executed when the program starts running. Its main purpose is to define the functionality of each pin and the initial state of the Arduino upon boot. For example, when creating a basic setup function with an ultrasonic sensor, a servo motor and a DC motor connected to an H-bridge, the function will look as follows:

```
void setup () {
  pinMode("trigPin number", OUTPUT);  // define the trigger pin as an output
  pinMode("echoPin number", INPUT);  //define the echo pin as an input
  pinMode("EnA number", OUTPUT);  //define the enable pin of the H-bridge as an output
  pinMode("In1 number", OUTPUT);  //define the In1 pin of the H-bridge as an output
  pinMode('In2 number", OUTPUT);  //define the 1n2 pin of the H-bridge as an output
  servo.attach("pin number");  // attaches the servo on to the chosen pin number
}
```

The second block, which is the execution block, is a loop function that hosts statements and conditions for reading inputs and triggering outputs, and executes them repeatedly. In other words, a loop in programming languages is a determined sequence of instructions which are continuously repeated until a certain condition is met or the program is shut down. This block will include the main instructions which will be controlling the car model.

The first lines of code which are needed are those which allow the sensors to begin measuring the distance and providing it as an input to the microcontroller. First, using the function "digitalWrite", the trigger pin of the sensor has to be set at high in order to start sending pulses of ultrasonic waves. After the waves bounce off an obstacle, they're picked up by the receiver (echo pin) and the total duration is measured using the "pulseIn" function. The next line of code is basically equation (2), which calculates the distance between the sensor and the obstacle using the time it took for the waves to travel back and the speed of sound.

In order to simplify the controlling of both the electric motor and the servo motor, a set of functions are created to be easily inserted into the driving conditions. Four functions are required to represent the four possible motion directions of the car (forward, backward, left, and right). By using the "digitalWrite" function, the forward motion can be achieved by sending a high signal to input 1 of the H-bridge while setting input 2 at low (reversing the signals produces motion in the backward direction). However, to produce motion in either the right or left direction, then an extra line of code has to be added which rotates the servo motor controlling the steering at a defined angle. For example, a function created for the purpose of steering the car in the right direction:

```
void turnRight() {
  servo.write(152);  //moves to the servo to 152 degree position
  digitalWrite(In1, HIGH);  //sends high signal to In1 pin
  digitalWrite(In2, LOW); //sends low signal to In2 pin
}
```

After introducing the distance as an input and creating the motion functions, conditions for the driving of the car can be written. An "if loop" is used here to determine the conditions under which the car shall be controlled. From programming we know that an "if loop" performs an action if a particular condition is satisfied. The first step begins by allowing the car to start moving forward if the sensor at the front of the car (Ultrasonic sensor 1) doesn't detect any obstacle at a distance less than 20 cm. However, should an obstacle suddenly appear at 20 cm or less than the front side of the car, then a decision has to be made whether the car should be turning left, right, or simply going back in reverse. In such a case the car will depend on the inputs the Arduino board receives from the three other sensors and produces an output to the servo motor/DC motor in order to determine the next movement the car has to perform. The function "delay()" is used after some conditions. The purpose of this function is to delay the time it takes the microcontroller to move on to the next line of code by a determined amount of milliseconds. The logic sequence of the code is visually represented in a flowchart for better understanding:

```
if (distance1 > 20) {
  servo.write(62);
  goForward();
 }
 else {
   if (distance2 < 20 && distance3 < 20 && distance4 > 20){
    Stop();
    servo.write(62);
    goBackward();
   delay(1500);
  }
  else if (distance2 > distance3){
   Stop();
   goBackward();
   delay(500);
   turnLight();
  delay(1500);
  }
  else if (distance3 > distance2){
   Stop();
  goBackward();
  delay(500);
  turnLight();
 delay(1500);
  }
```

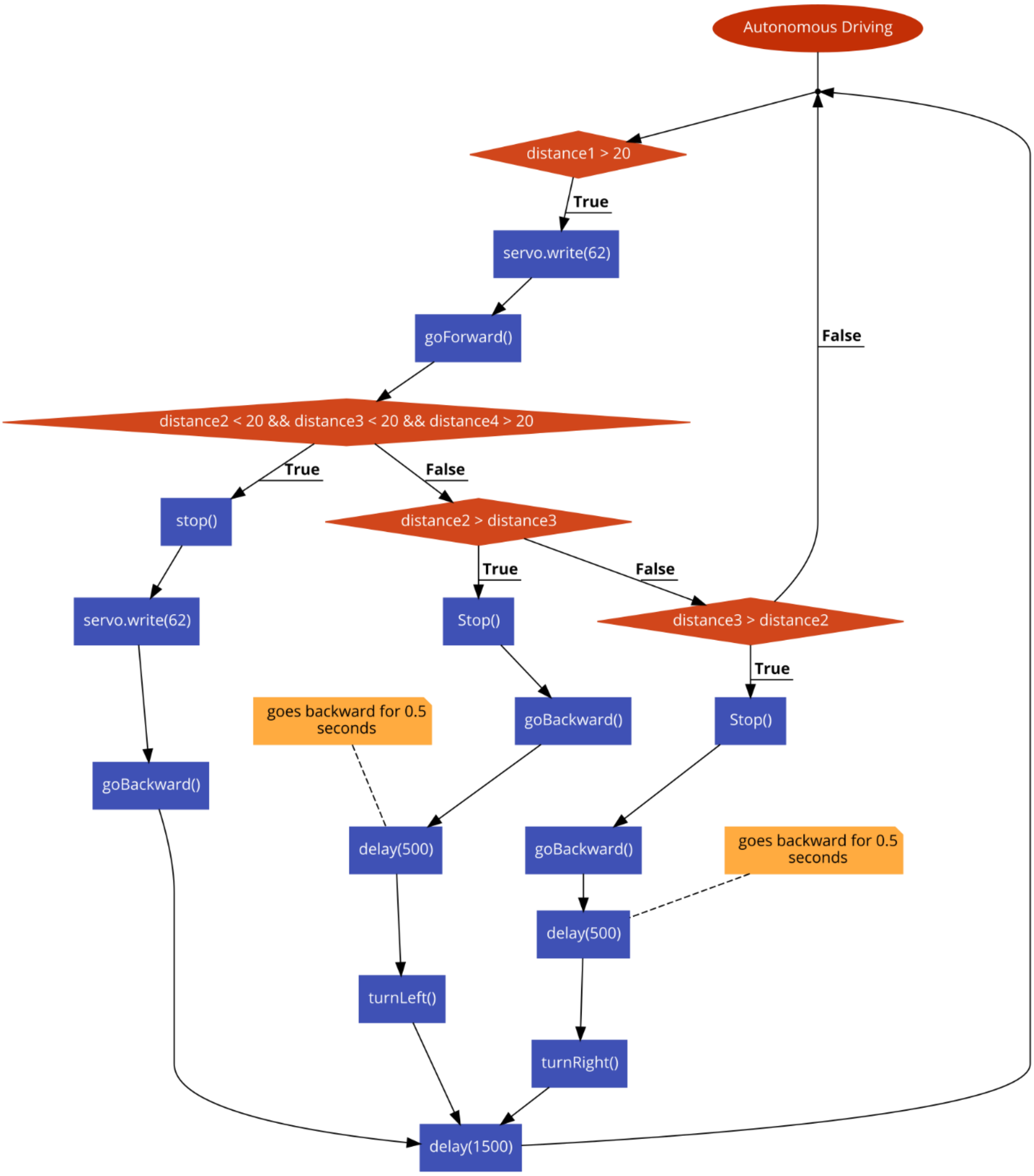

Figure 14 A flowchart representing the logic sequence of the code

Chapter 3

3.1 Testing and Experimentation

As we arrive to the final chapter, this experiment will be concluded by a series of five test drives to evaluate the functionality and the limitations of the car model. A maze was designed and built to ensure the car is always moving in the same environment and is subjected to the same conditions. Thus, these conditions will make the results of the tests easier to compare. For each test, the readings from each sensor, as well as the position of the servo, at every instant was recorded. The results which will be discussed are exclusive to the variation of the distance between the front side of the car (distance1) and the walls of the maze as the car is moving, and how the servo motor reacts and changes position with time. The collected data will be plotted in two separate charts. The first chart which represents the readings of the front sensor as a function of time will include a dotted line indicating the distance of 20 cm as the threshold distance at which the servo begins to change position from the initial forward position whenever the measured distance falls below 20 cm. In the second chart, the position of the servo motor is plotted as a function of time with a dotted line indicating the initial forward position at 62°. Arrows are used to link the change in the position of the servo motor to the change in distance in the first chart whenever the distance drops below 20 cm. Even though the car is expected to complete the maze in every test drive, results are expected to vary between one test and another. Thus, each set of data from every test will be discussed thoroughly in each of the following subchapters. The aim of repeating the same test five times is to be able to see whether there will be large changes in the results caused by small variations in the lab such as small deviations in the starting position or angle of the car upon entering the maze.

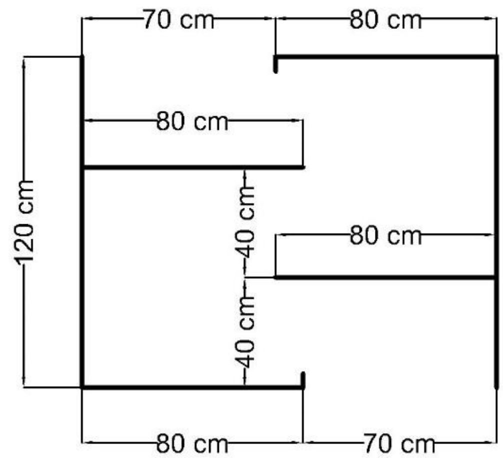

Figure 15 A Sketch of the Maze Where the Testing Occurs

3.1.1 The First Autonomous Test

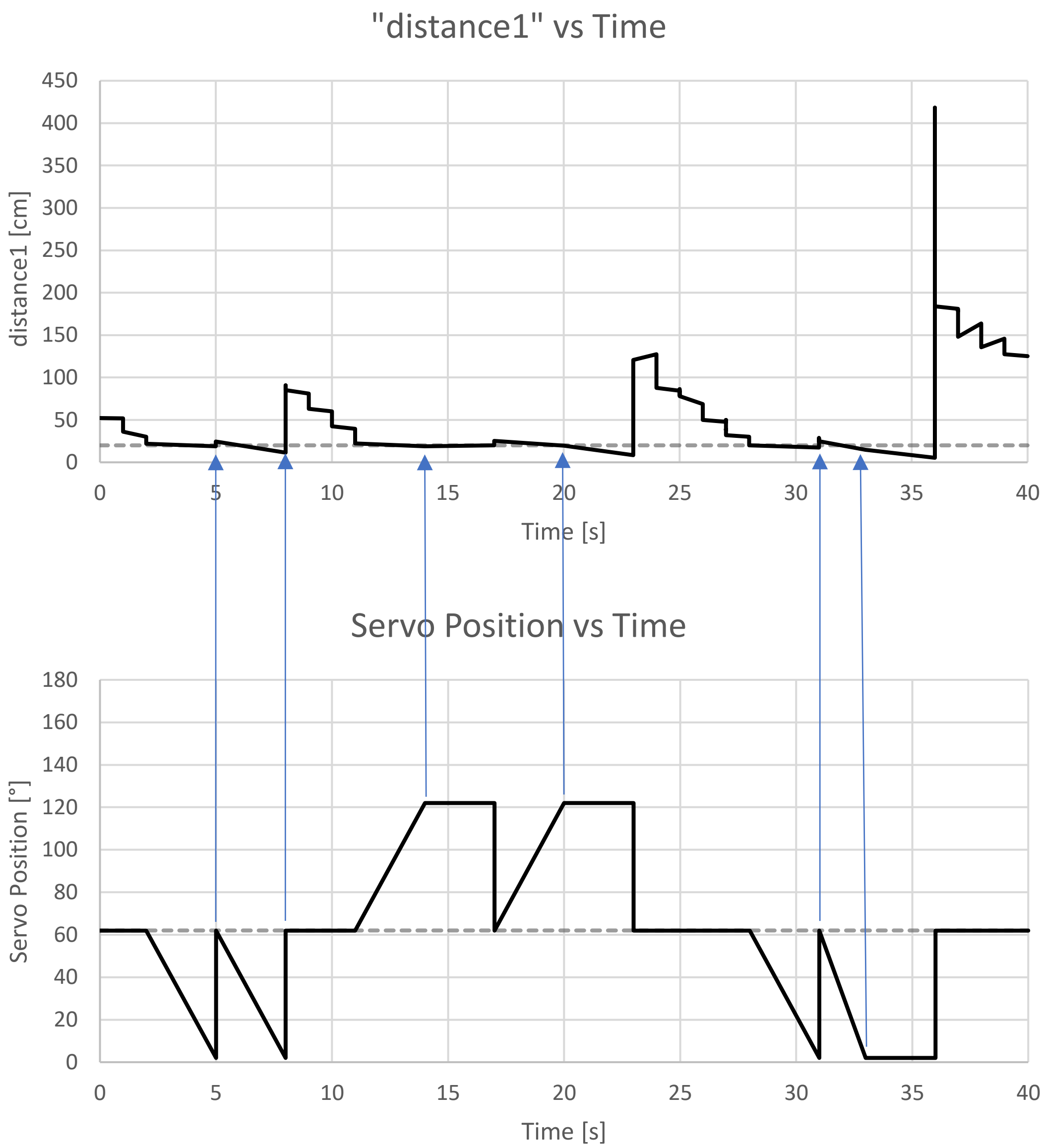

Figure 16 Results of the first autonomous test

In the first test, it took the car only a remarkable 40 seconds to complete the maze. From the graphs we can observe that the car had minimal difficulties while navigating through the maze. From the recorded data, the first reading of the front sensor shows a distance of 52.2 cm from the wall of the maze. As the car enters the maze, this distance soon decreases below the threshold till it reaches 18.95 cm which produces a reaction in the servo motor at the time of 5 seconds. This reaction is dependent on the readings of the other sensors. At this instant distance2 is 127.51 cm and greater than distance3 which is 30.6 cm, thus the car turns left and the servo motor switches from 62° to 2°. This reaction is repeated three seconds later when the distance immediately falls below the threshold again to 11.34 cm. This time distance2 was 58.4 cm while distance3 was 6.55 cm, thus the car turns left again. We can observe an increase in “distance1”, as it peaks at 90.74 cm, which means that the car has managed to avoid the wall of the maze and continue on its path by turning left. At time of 14 seconds, another reactions occurs in the servo motor. At this instant, the distance at front sensor drops to 18.85 cm while the distance on the left side is 22.86 cm which is less than the distance on the right side that is 58.19 cm, thus the angle of the servo motor increases from 62° till 122° and the car turns right. This reaction produces a response that lasts three seconds longer, because it takes the car until time reaches 17 seconds for the front sensor to rise above the threshold of 20 cm. Three seconds later and the same reaction is repeated as “distance1” drops till 19.47 cm and continues decreasing for three more seconds until it reaches 8.33 cm, while the distance on the right side remains larger than the distance on the left side. At this stage, the car has completely managed to navigate around the left side of the maze and is now traveling in a straight line through the middle part of the maze. At the time of 31 seconds, the car has gotten very close to the right wall of the maze and the distance at front of the car is 17.51 cm. The car will now have to turn left and thus the servo motor changing from 62° to 2°, since the left distance, which is 34.8 cm, is greater than the right distance, which is 23.15 cm. The response this time was short and not sufficient for the car to completely avoid the right wall, for that reason another response is observed only two seconds later as the front distance decreases again below the threshold till it reaches 14.77 cm and continues to decrease for three more seconds till 5.32 cm. Now we can observe a longer reaction from the servo which lasts for 3 seconds and proves to sufficient for the car to completely navigate away from the wall. As the car is moving now inside the maze, its trajectory has become a bit skewed. This causes the car to manage to exit the maze directly in an irregular fashion without needing to navigate around the left wall of the maze. The huge rise in “distance1”, reaching 418.2 cm, which we observe in the graph is due to the car exiting the maze and the sensor detecting the distance to an object inside the room where the maze was placed. The ending was an unplanned event but the car was able to fulfill its purpose regardless without by navigating through the whole maze without crashing.

3.1.2 The Second Autonomous Test

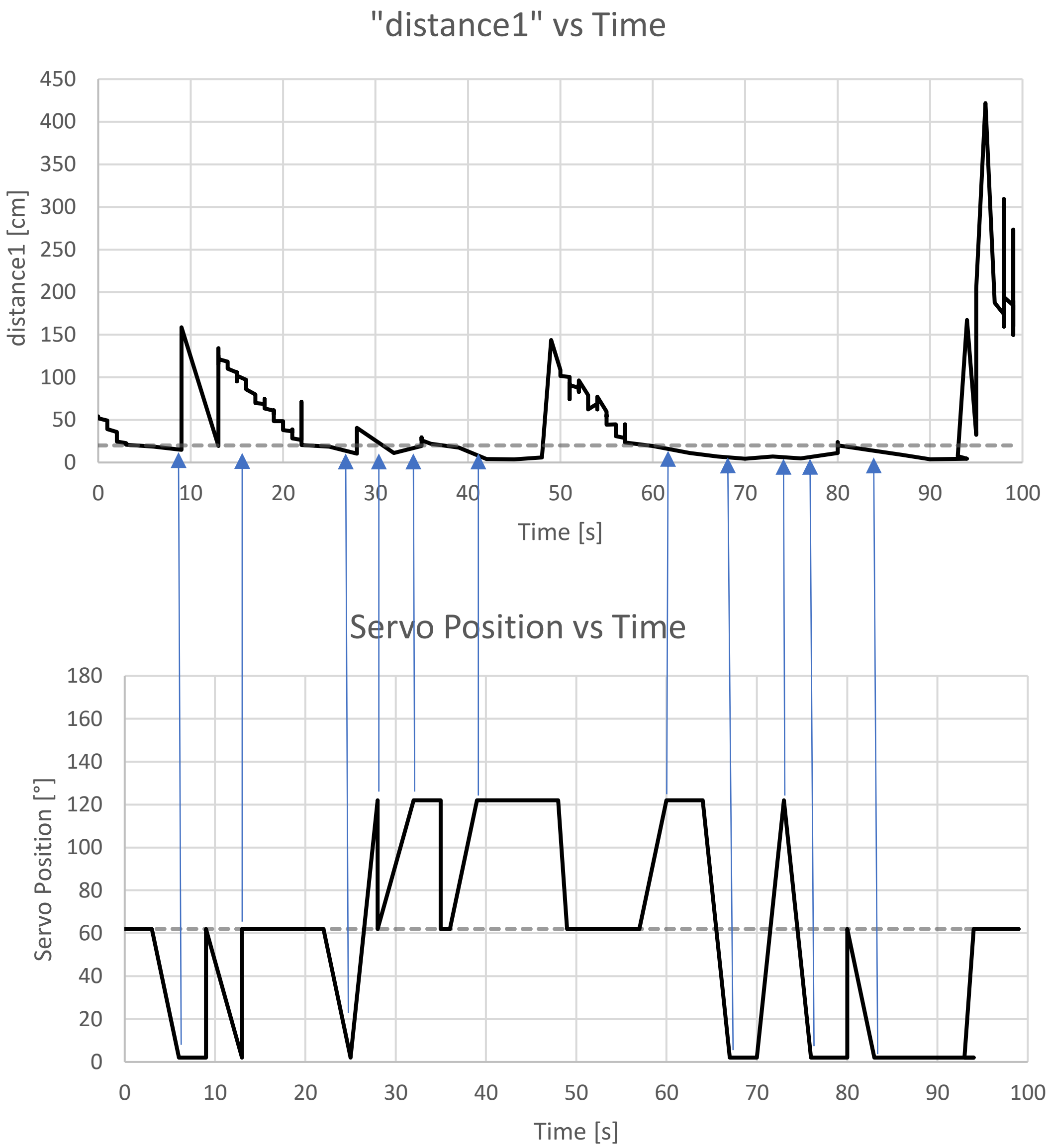

Figure 17 Results of the second autonomous test

The second test took more time than first one and was completed after 99 seconds. From observing the graphs we can see that the test drive was not as smooth as the previous test. The first recorded reading of the front sensor indicates a distance of 53.95 cm from the inner wall of the maze. As the car begins to enter the maze, this distance drops till 18.74 cm after 6 seconds, and thus is now below the threshold. At this instant, the left sensor reads 158.89 cm while the right sensor reads 10.75 cm. Since the left distance measured is greater than the right one, then the car is expected to turn left and the servo position changes from 62° till 2°. Since "distance1" remains below the threshold until time is 9 seconds, then the reaction in the servo lasts for 3 seconds. Four seconds later, the front distance drops below the threshold again till it reaches 19.31 cm and the same reaction is repeated by the servo motor, considering that the left distance is still greater than the right distance. Now the car has completely avoided the first inner wall of the maze and begins to move forward to the left side of the maze. When the time reaches 25 seconds, the front distance drops till 18.61 cm as it reaches the left wall of the maze. However, we can observe a very unusual reaction in this step since the car has turned left instead of turning right. The recorded measurements of the left distance and the right distance are 41.76 cm and 33.79 cm respectively, which shows why the servo motor has reacted in this manner. The reason for this unexpected event can be attributed to the car deviating slightly from its course after successfully avoiding the first obstacle and thus reaching the second obstacle in a slightly skewed position. Three seconds later, as the left distance (31.5 cm) becomes less than the right distance (37.87 cm), then the car begins to turn right and the servo position changes to 122°. As the front distance drops to 11.18 cm when the time reaches 32 seconds and remains below the threshold for the next 3 seconds, we can observe that the car turns right again. At this moment the car still has yet to completely avoid the inner wall of the maze and continue on its course. This action is finally achieved after 4 seconds when the front distance drops till 17.42 cm and continues to decrease till it reaches 5.99 cm, which produces a longer reaction in the servo motor and thus the car managing to avoid the wall and proceed in the maze. When the time reaches 60 seconds, we see the same event from the time of 25 seconds reoccurring except now the car has turned to the right for the same reason as discussed before. Three seconds later, the car turns left only to begin turning right again after 6 seconds. Here we are seeing the car struggling to adjust as the right distance increased massively from 14.08 cm to 99.3 cm, thus causing the left distance to change from being greater than the right distance to suddenly becoming lower. After three seconds have passed, the car has regained its normal movement and manages to turn left and avoid crashing into the right wall of the maze. As the car moves forward now and due to its slightly skewed position, it manages to successfully exit the maze at time of 99 seconds in the same manner the previous test and without needing to avoid the left wall of the maze. In the graph, again we can see huge rises in “distance1” which is caused by the front sensor detecting objects in the room where the maze was placed as the car exits.

3.1.3 The Third Autonomous Test

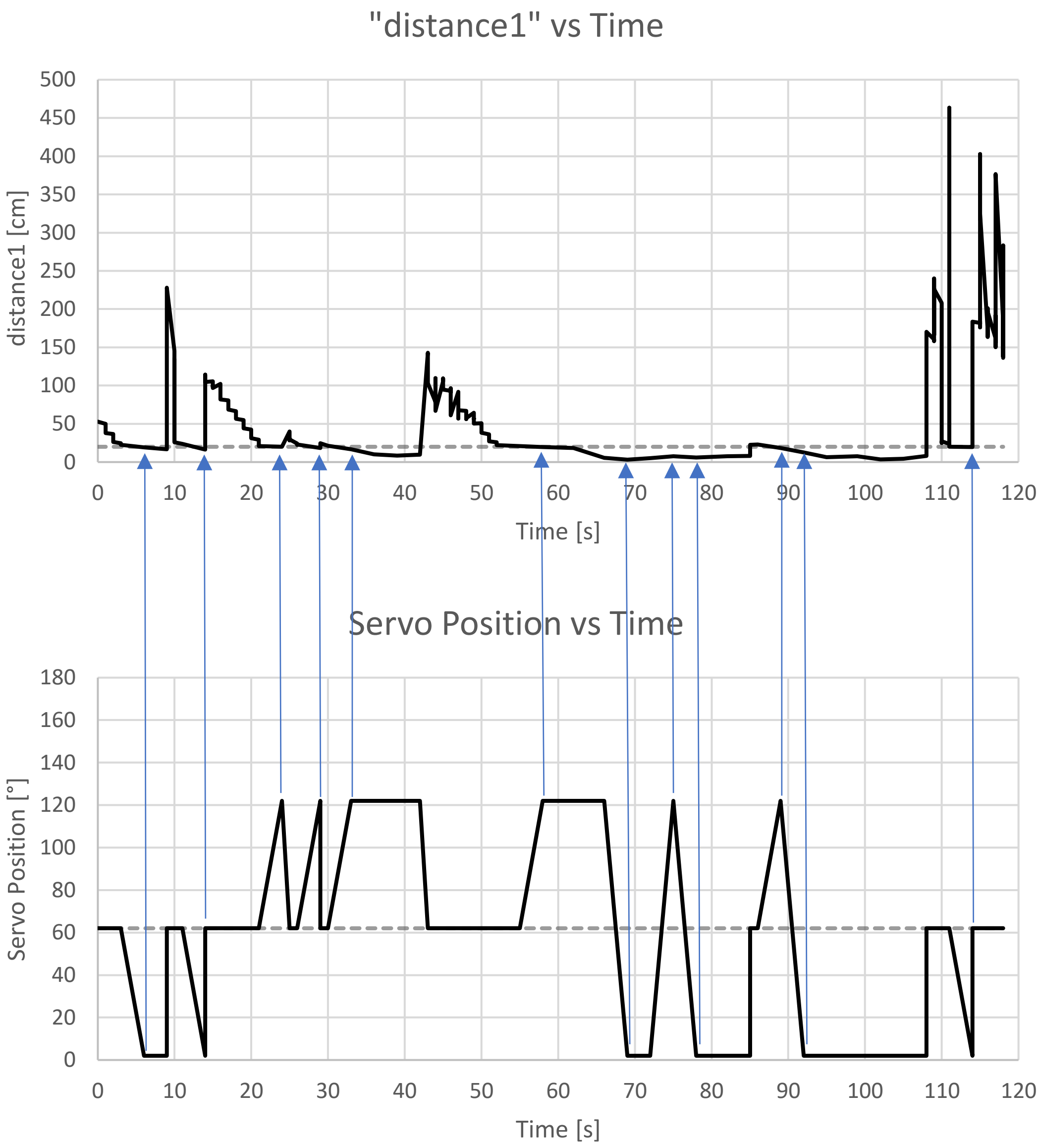

Figure 18 Results of the third autonomous test

The third test is even longer than both the first and the second test, since it took the car 118 seconds to finish the maze. We see the car entering the maze with a recorded distance from the sensor of 52.75 cm. As the car begins approaching the inner wall of the maze, the first reaction from the servo motor is provoked at a distance of 19.36 cm at a time of 6 seconds and lasts for 3 seconds. In this instant, the left distance was 152.36 cm while the right distance was 14.87 cm, thus as expected the servo position changes from 62° till 2° and the car turns right for 3 seconds and then continues to move forward. However this action was not enough to completely avoid the wall and the front distance drops below the threshold again 5 seconds later to reach 16.52 cm. With the left distance (201.6 cm) remaining much larger than the right distance (10.32 cm) then the servo positon changes to 2° again for a second. Now the car has completely avoided its first obstacle inside the maze and continues to move forward to the left side. After 10 seconds of moving forward at a servo position of 62°, the car eventually meets a new obstacle in the form of the left wall of the maze. At time of 24 seconds, the front sensor of the car records a distance of 19.93 cm, while the left distance (29.31 cm) is less than the right distance (49.94 cm) which leads to the servo position to change from 62° till 122° and force the car to turn right. The same action is repeated after 5 seconds as the car attempts to navigate around the left wall of the maze. At a time of 33 seconds, the front distance drops below the threshold again at 16.94 cm and continues to decrease and remain below the threshold for 9 more seconds. During all these 9 seconds, the left distance remains much lower than the right distance which causes the car to turn right and the servo position to change to 122°. Here we are observing the car making one strong push in order to completely avoid crashing into the inner wall of the maze and continue on its trajectory. After having successfully avoided crashing into any obstacles or walls in the maze, the car continues in its forward trajectory for 15 seconds. The front distance of the car drops below the threshold once again now at 19.53 cm when it reaches the right wall of the maze. At this instant, the car begins to suffer great difficulties in its ability to navigate. The car begins to turn right instead of turning left and continues to get closer to the right wall. This action, which lasts for 7 seconds, is caused by the slightly skewed trajectory of the car that is cause by its previous obstacle avoidance. The car immediately responds to the reading from the sensors now as it's about to crash and the left distance (32.4 cm) becomes larger than the right distance (9.19 cm), thus the car turns left now for 3 seconds and avoids collision with the wall. After 4 seconds, we see the car struggling again as it turns to the right when the left distance (6.35 cm) becomes lower than the right distance (11.7 cm) once again. However 3 seconds later the car manages to correct its previous action and turn left as the left distance rises till 8.52 cm and the right distance drop till 3.52cm. As the car keeps moving forward now for the next 3 seconds, it reaches the last wall of the maze and the servo position has to change one last time to 2° to enable the car to avoid collision with this wall. Now the car is able to keep moving in a forward manner with a servo position at 62° to exit the maze without any more complications at a time of 118 seconds.

3.1.4 The Fourth Autonomous Test

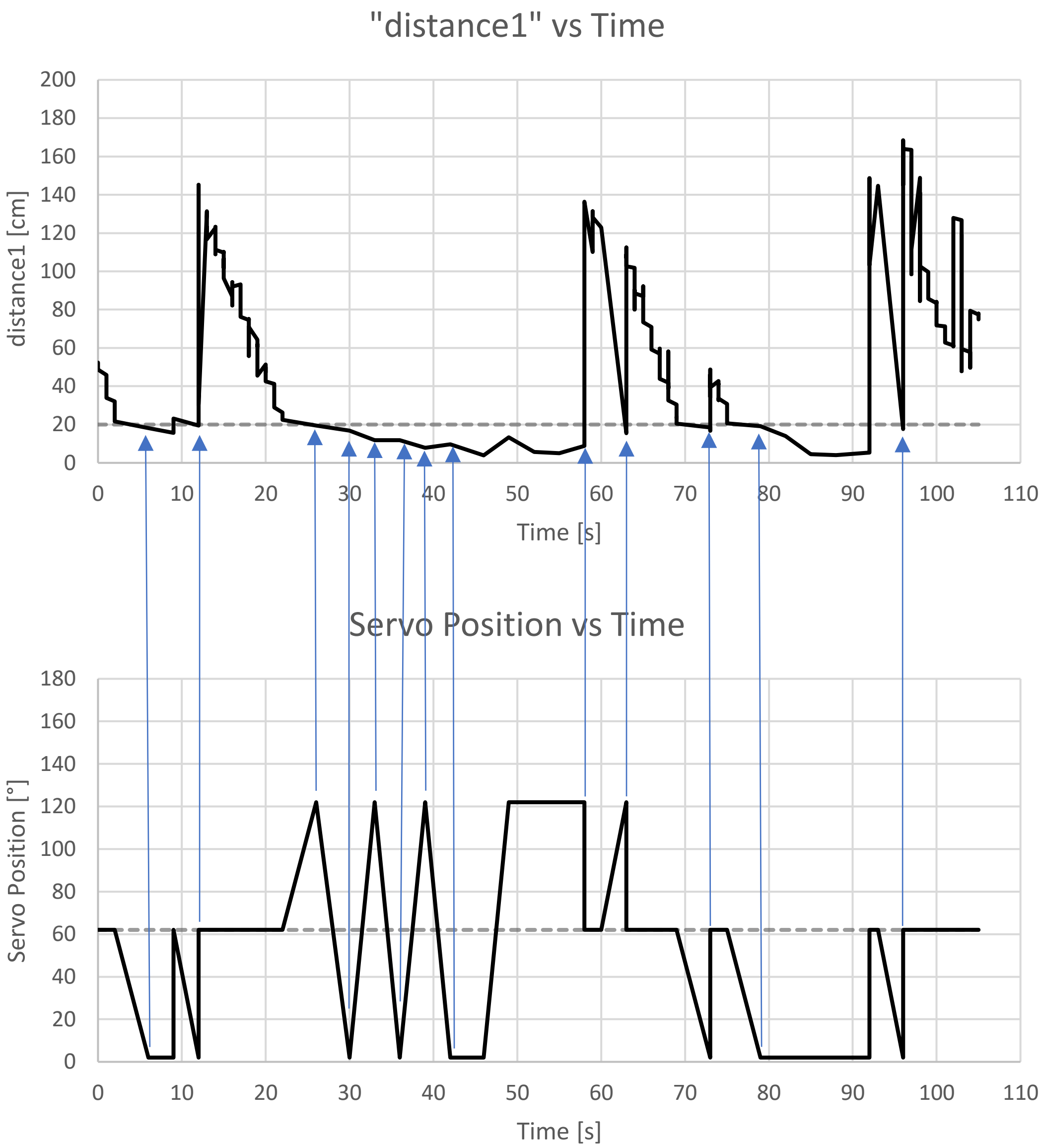

Figure 19 Results of the fourth autonomous test

After the first three tests, we are able now to predict a pattern or a series of events of what wanted and unwanted actions the car might do. In the fourth test, as the car enters the maze, the recorded distance by the front sensor was 52.41 cm at a servo position of 62°. After 6 seconds, the car approaches the inner wall and the maze and “distance1” drops below the threshold for 3 seconds to reach 15.68 cm. As the left distance is recorded as 158.23 cm and is greater than the right distance which is 16.34 cm, then the servo position changes to 2° and the car turns left. We see this action repeated at a time of 12 seconds when the front distance falls to 19.41 cm. With the car having successfully avoided crashing into the first inner wall of the maze, it starts to move forward to the right for the next 14 seconds. Now that the car has arrived at a distance of 19.45 cm from the left wall of the maze, and since the left distance (3.17 cm) is much lower than the right distance (13.98 cm), then the servo position changes to 122° and the car turns right. For the next 20 seconds the movement of the car becomes very chaotic as it fluctuates between turning left and turning right. We can see in the graph a huge variation of the servo position between the time interval ranging from 26 seconds to 46 seconds. At this time instance, the car becomes trapped somehow in the left corner of the maze with the left and right distances changing constantly and thus causing a constant variation in the servo position. As the car attempts to free itself from this trap, it finally manages to proceed with its expected trajectory at a time of 49 seconds. Now the left distance becomes 15.13 cm and is lower than the right distance, which is 15.16 cm, by a very small margin. The servo position now changes to 122° and the car turns right. As the car is turning right, the left distance continues to decrease while the right distance continues to increase, thus the car continues to turn right for the next 9 seconds. The car now proceeds to move forward for 5 seconds until it gets very close to the inner wall of the maze at a distance of 15.45 cm. Since the left distance (4.7 cm) is much lower than the right distance (137.06 cm), the servo position changes to 122° and the car has finally navigated around the inner wall of the maze and is free to continue moving forward. After 10 seconds, the car reaches the right wall of the maze and the front distance is recorded at 18.56 cm. Now the left distance is 51.42 cm and much great than the right distance which is 5.93 cm, thus the car turns to the left as the servo changes position to 2°. The car has now progressed to the final wall of the maze and 6 seconds later it arrives to the wall at a front distance of 19.16 cm. The servo motor's reaction is repeated once again as the left distance (170.27 cm) remains greater than the right distance (8.95 cm). After 4 seconds, the car will have to make one last adjustment to its trajectory, as front distance drops below the threshold one last time to reach 17.68 cm and thus causing another reaction in the servo motor to change the position to 2° and turn left. Now the car is able to continue moving forward and exit the maze successfully.

3.1.5 The Fifth Autonomous Test

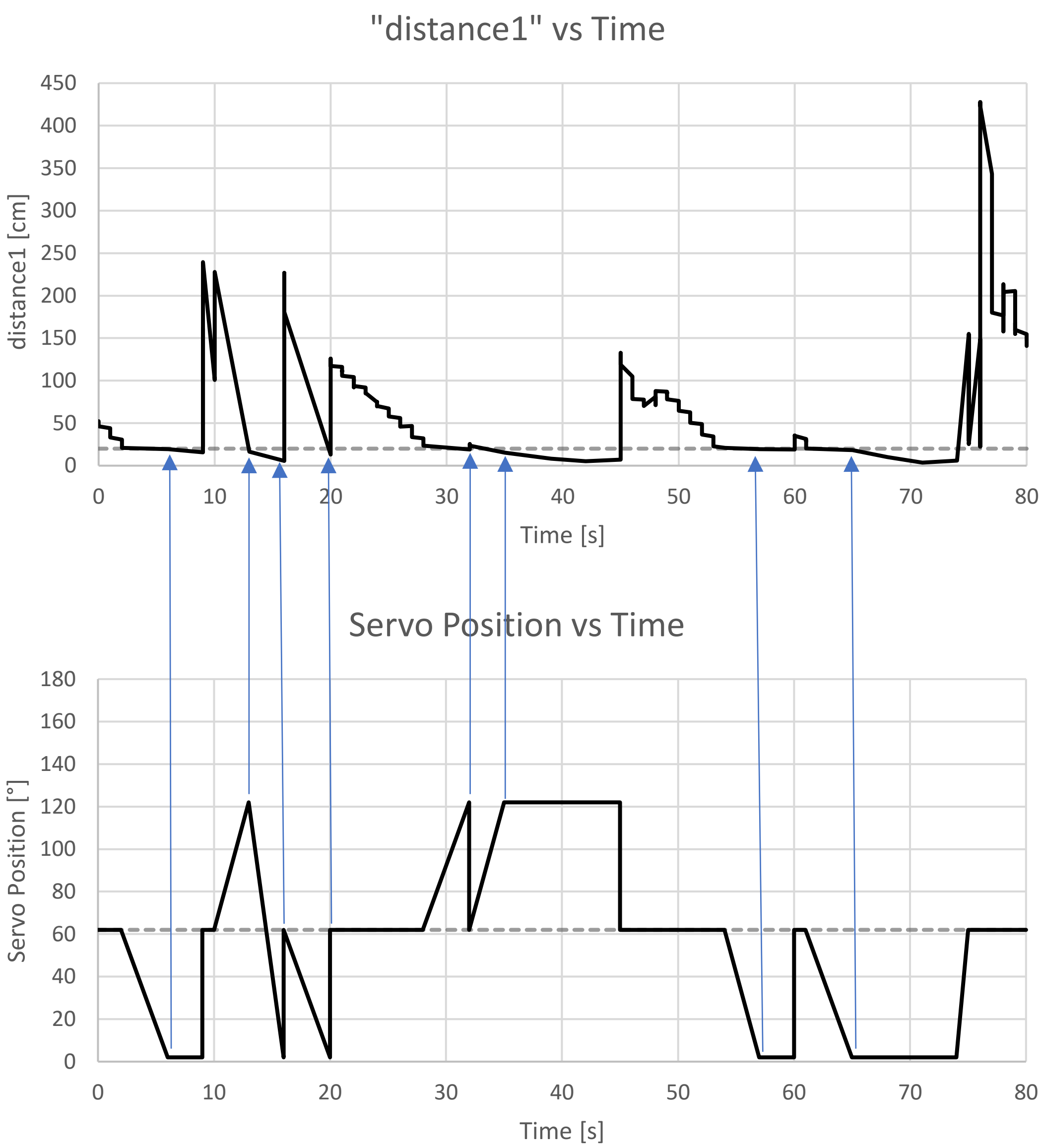

Figure 20 Results of the fifth autonomous test

As we reach the fifth and final test, we see that the car manages to complete the maze in 80 seconds, which is the second lowest time needed after the first test. The car starts by approaching the maze at a distance of 52.24 cm from the maze's inner wall. After 6 seconds have passed, the car is separated by a distance of 19.52 cm from the wall and continues to decrease till 15.59 cm at time of 9 seconds. Since the front distance is now below the threshold distance, the servo motor will have to react and move from 62° to 2° after detecting the left distance (164.88 cm) as higher than the right distance (18.88 cm). After 4 seconds, the car unusually turns right before even avoiding the inner wall. At this instant, the left sensor recorded 75.55 cm while the right sensor recorded 76.08 cm. We can observe the small margins between the left distance and the right distance which occur while the car is on the turn inside the first corner of the maze. The small margins occurring at an instant are able to completely take the car off its course. After 3 seconds, the sensors respond to the car's irregular movement and help send it back on its path by triggering a reaction in the servo motor to turn left, since the left distance (111.27 cm) began to increase while the right distance (34.95 cm) decreased noticeably. The car now will have to make one last left turn in order to avoid collision with the wall. This action is evident after 4 seconds as the left distance (129.47 cm) continues to rise much higher than the right distance (4.49 cm). At the time of 32 seconds, the car has finally reached the left wall of the maze and the front distance becomes 19.23 cm. Now the car performs a right turn when the servo move to 122° as the left distance is 5.37 cm and much lower than the right distance, which is 27.87 cm. This action is repeated 3 seconds later, when the car performs a right turn for a longer period of time lasting 10 seconds and completely managing to successfully navigate around the corner. It's clear now that after suffering a slight difficulty in the beginning of the maze, the car goes on a very smooth and non-challenging path which is evident in the graphs. The car now reaches the right wall of the maze and the front distance drops till 19.57 cm when the time reaches 57 seconds. The left distance (43.51 cm) is now much higher than the right distance (14.1 cm), thus the servo position changes to 2°. After 5 seconds, the same action is repeated for a longer period of time. Here we see the car attempting to navigate around the last wall of the maze by turning left for 9 seconds. The car is now able to move in a straight manner and exit the maze from a tight angle instead of having to turn away from the left wall near the exit. This test would have been almost faultless if it not had been to a small setback in the beginning of the maze.

After completing the fifth test, we can observe the variation in the time it has taken for the model to complete the maze across all the tests done. This variation will be represented appropriately in the following table:

Test	Time [s]	Mean Average [s]	Standard Deviation [s]
1	40	88.4	27.12
2	99		
3	118		
4	105		
5	80		

Table 10 Summary of results of the autonomous tests

Chapter 4

4.1 Conclusion

This thesis has examined the possibility of developing an autonomous model and the challenges associated with such a project. The initial plan was to produce two functional models of the car using a Lidar sensor and an ultrasonic sensor. Unfortunately, the model using the Lidar technology was not built due to having a limited budget and the damaging of the only available Lidar sensor in our laboratory by overvoltage. However, using four ultrasonic sensors, a model was successfully made and tested.

The approach to develop this project was made with the help of an Arduino microcontroller board where all key components for controlling the car's movement, as well the sensors, were directly connected to the board in order to create a combined network of inputs and outputs between them. The servo motor was used to control the steering of the front wheels of the car while the DC geared motor was responsible for the forward and backward rotation of the rear wheels. The L298n H-bridge played a role in controlling the DC geared with the Arduino board. However, the main part of this work was connecting the ultrasonic sensors to the board and producing inputs to force a reaction from the components. These so called reactions are what generate the action of autonomous driving. Basically, the sensors read the distance between different sides of the car and obstacles, and then input these distances into the Arduino board. A code is developed and uploaded into the board which makes sense of these inputs and then runs them in a loop containing conditions for the behavior of the components. Depending on the values of the inputs, or in other words the distances measured by the sensors, the code checks which conditions are satisfied and then proceeds to send signals to the components. The behavior of the components then reflects the position of the car with respect to the surrounding obstacles and define its movement.

The concept of the maze was to give the car model a predetermined test track and allow for observation and analysis of its logic. In this thesis, there was five documented and thoroughly analyzed test drives in this maze. The results of these tests proved the success of this model in terms of navigating autonomously inside a maze and managing to reach the exit. However, due to the limitations of the sensors and the other components, the ride was not as smooth as expected. In most of the test results, we are able to observe moments where the readings of the sensors managed to confuse the car and send it off its expected track before finally managing to adjust after a certain period of time. As both left and right sensors read measurements simultaneously, a confusion caused by the left and right steering conditions occur. A main reason which can be attributed to the confusions is that while the car is able to navigate left and right to avoid obstacles, it doesn't have a sense of position on a defined track or road and thus doesn't always travel in a straight manner inside the maze which causes it to reach corners and confuse between inputs. Another reason is the sensors ability to measure distances linearly only. In other words, when measuring distances to flat surfaces at an angle, the waves which are supposed to bounce back and be picked up by the receiver are mostly reflected into another trajectory at an angle equaling the angle of incidence.

4.2 Future Work and Propositions

Despite the overall success of this project, each part of the system could be improved. As it stands, the system represents the bare minimum functionality expected of an autonomous vehicle. One improvement can be done using a Lidar sensor and a software to map the environment and give the car a complete and accurate idea of all of its surroundings. However such a task requires a controller with more processing power than the Arduino board. Additionally, it would allow the use of sensors to be extended into detecting dynamic objects as well as static ones. This is fundamentally important if a functional autonomous car is to be produced for the consumer in terms of being able to detect and interact with surrounding vehicles in traffic as well as with the public. Perhaps the use of a variation of sensors is the key for the future of the autonomous vehicle, where these different types of sensors are able to complement each other in the areas of weakness or lower performance. The most developed vehicle now a days in this field is the "Tesla Autopilot" which has the most advanced driver-assistance system containing lane centering to keep the car centered in the lane , adaptive cruise control which automatically adjusts the vehicles speed to maintain a safe distance from other vehicles ahead, self-parking and the ability to switch lanes on the highway. [36] However, a lot of controversy has surrounded this innovative vehicle since it has led to six well known accidents where two of them caused the death of the car's drivers.

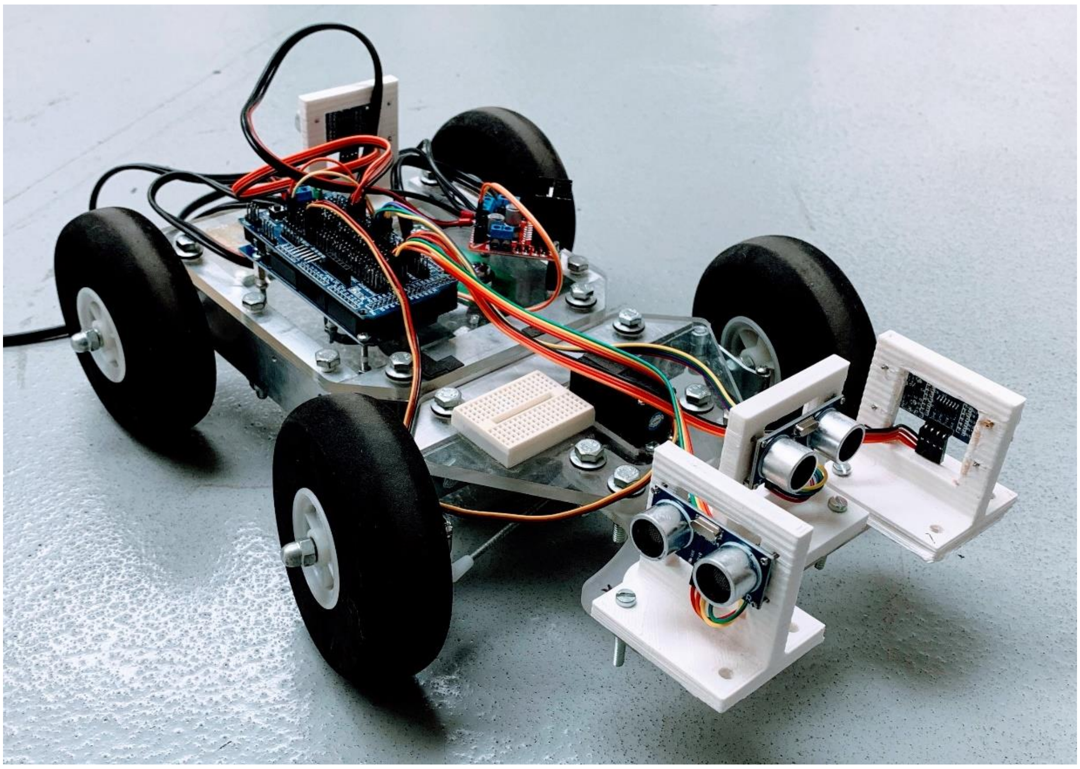

Figure 21 The car model with the ultrasonic sensor

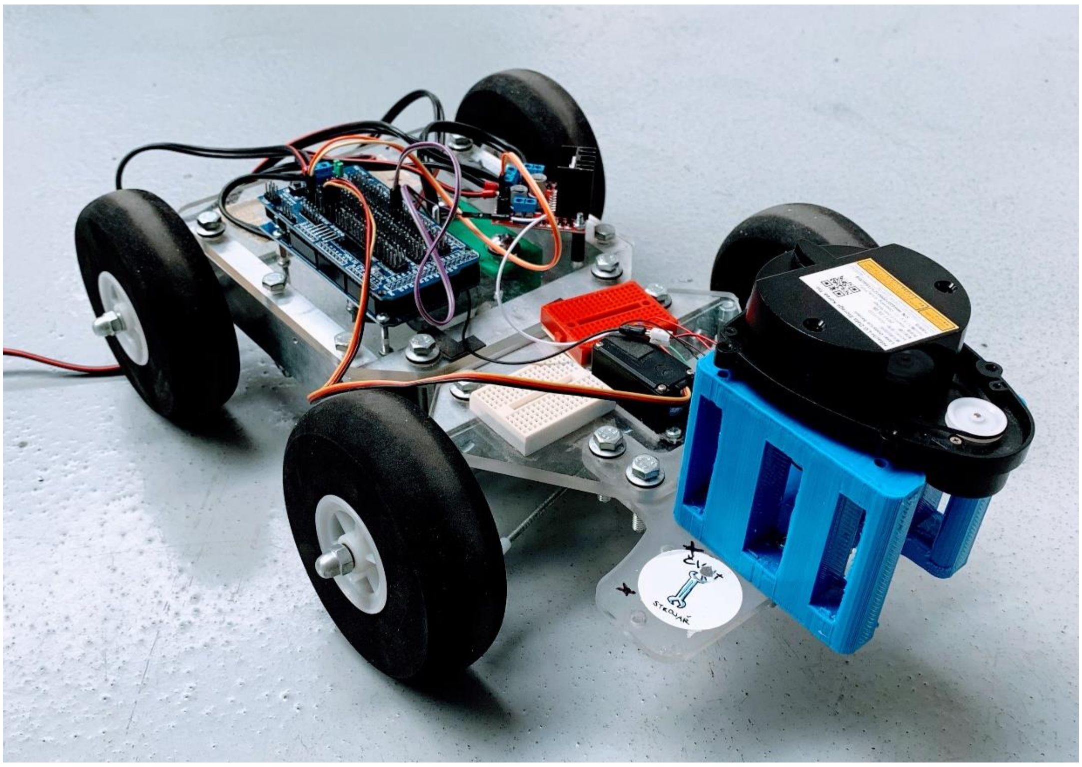

Figure 22 The car model with the Lidar sensor

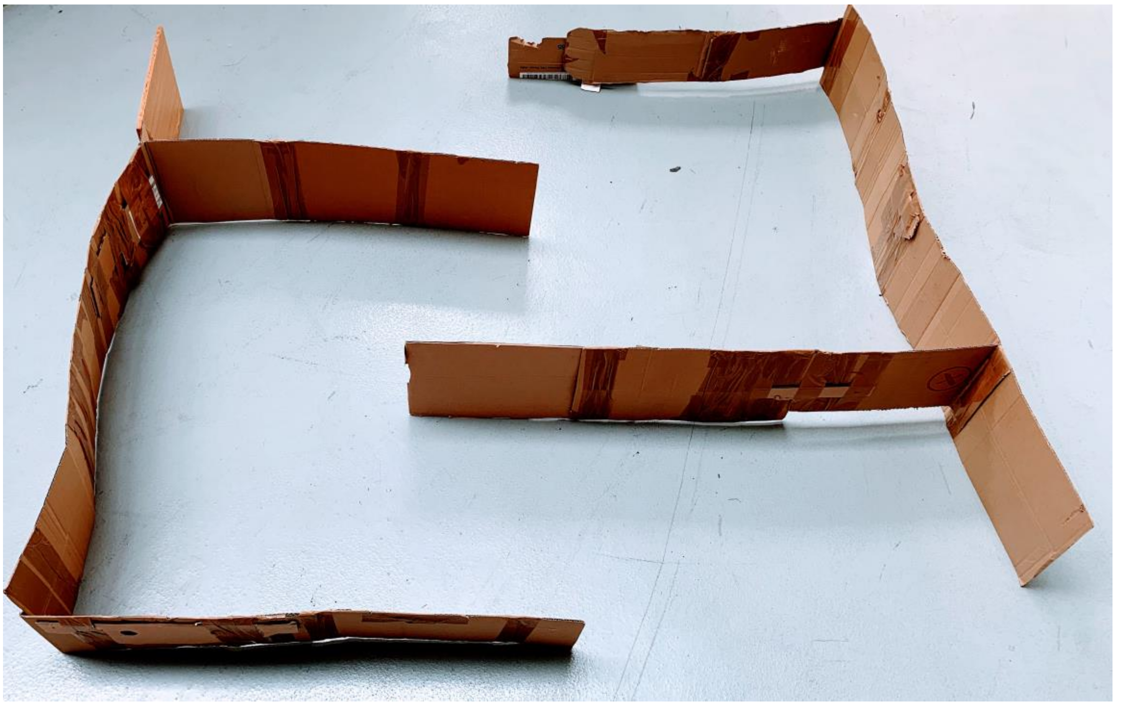

Figure 23 A cardboard model of the maze

References

[1] "Time Magazine. Science: Radio Auto [online]. [cit. 2019-03-16]. Time Magazine,1925, Aug. 10, Vol. (6). Available from: http://content.time.com/time/magazine/article/0,9171,720720,00.html," [Online].

[2] "GRINGER, Bonnie. History of The Autonomous Car. TitleMax [online]. [cit. 2019-03-16]. Available from: https://www.bosch-mobility-solutions.com/en/products-and-services/passenger-cars-and-light-commercial-vehicles/driver-assistance-systems/construction," [Online].

[3] "What Is Lidar, Why Do Self-Driving Cars Need It, And Can It See Nerf Bullets ?. Wired [online]. 2018 [cit. 2019-06-03]. Available from: https://www.wired.com/story/lidar-self-driving-cars-luminar-video/," [Online].

[4] "LEVIN, Sam. Uber crash shows 'catastrophic failure' of self-driving technology. The Guardian [online]. [cit. 2019-03-16]. Available from: https://www.theguardian.com/technology/2018/mar/22/self-driving-car-uber-death-woman-failure-fatal-crash-arizona," [Online].

[5] "YDLIDAR X4 360 ° Laser Scanner. Robot Shop [online]. [cit. 2019-03-16]. Available from: https://www.robotshop.com/eu/en/ydlidar-x4-360-laser-scanner.html," [Online].

[6] "YDLIDAR G4 360 ° Laser Scanner. Robot Shop [online]. [cit. 2019-03-16]. Available from: https://www.robotshop.com/en/ydlidar-g4-360-laser-scanner.html," [Online].

[7] "RPLIDAR A2M8 360 ° Laser Scanner. Robot Shop [online]. [cit. 2019-03-16]. Available from: https://www.robotshop.com/en/rplidar-a2m8-360-laser-scanner.html," [Online].

[8] "Christian Huelsmeyer, The Inventor. Radar World [online]. 2007 [cit. 2019-06-03]. Available from: https://www.radarworld.org/huelsmeyer.html," [Online].

[9] "PICKERING, Paul. Radar Technology Behind Autonomous Vehicles. ECN Mag [online]. [cit. 2019-03-16]. Available from: https://www.ecnmag.com/article/2017/12/radar-technology-behind-autonomous-vehicles," [Online].

[10] "The Fundamentals of Millimeter Wave Sensors. Texas Instruments [online]. [cit. 2019-05-22]. Available from: http://www.ti.com/lit/wp/spyy005/spyy005.pdf," [Online].

[11] "Texas Instruments AWR1243BOOST. Mouser [online]. [cit. 2019-03-16]. Available from: https://eu.mouser.com/ProductDetail/Texas-Instruments/AWR1243BOOST," [Online].

[12] "Texas Instruments AWR1443BOOST. Mouser [online]. [cit. 2019-03-16]. Available from: https://eu.mouser.com/ProductDetail/Texas-Instruments/AWR1443BOOST," [Online].

[13] "Texas Instruments AWR1642BOOST. Mouser [online]. [cit. 2019-03-16]. Available from: https://eu.mouser.com/ProductDetail/Texas-Instruments/AWR1642BOOST," [Online].

[14] A. Thakur, "THAKUR, Anshul. Engineers Garage. Ultrasonic (UV) Sensors or Ultrasound Sensors [online]. [cit. 2019-03-16]. Available from: https://www.engineersgarage.com/articles/ultrasonic-sensors," [Online].

[15] Bosch, "Ultrasonic Sensor. Bosch [online]. [cit. 2019-03-16]. Available from: https://www.bosch-mobility-solutions.com/en/products-and-services/passenger-cars-and-light-commercial-vehicles/driver-assistance-systems/construction-zone-assist/ultrasonic -sensor," [Online].

[16] "Devantech SRF05 Ultrasonic Range Finder. Robot Shop [online]. [cit. 2019-03-16]. Available from: https://www.robotshop.com/eu/en/devantech-ultrasonic-range-finder-srf05.html," [Online].

[17] "HC-SR04 Ultrasonic Range Finder. Robot Shop [online]. [cit. 2019-03-16]. Available from: https://wwwrobotshop.com/en/hc-sr04-ultrasonic-range-finder-osepp.html," [Online].

[18] "Weatherproof Ultrasonic Sensor. Robot Shop [online]. [cit. 2019-03-16]. Available from: https://www.robotshop.com/eu/en/weatherproof-ultrasonic-sensor-separate-probe.html," [Online].

[19] GIBSON, James Jerome. The Perception of the Visual World . Houghton Mifflin, 1950. ISBN 978-1114828087..

[20] "The Computation of Optical Flow . New York, USA: ACM Computing Surveys, 1995, (Volume 3).".

[21] MOINI, ALIREZA. Vision Chips . Illustrated. Springer Science & Business Media, 2012. ISBN 9781461552673-7..

[22] FLOREANO, Daria. Flying Insects and Robots . Illustrated. Springer Science & Business Media, 2009. ISBN 9783540893936..

[23] "Optical Flow Sensor APM2.5. Arduino Shop [online]. [cit. 2019-03-16]. Available from: https://arduino-shop.cz/arduino/1383-optical-flow-sensor-apm2-5-multicopter-adns-3080-opticky-senzor-pro-arduino.html," [Online].

[24] "PX4FLOW Camera Module. Amazon [online]. [cit. 2019-03-16]. Available from: https://www.amazon.com/PX4FLOW-Module-Camera-Without-SINGLE/dp/B07D4HNMB8," [Online].

[25] "ArduCam OV5647 Camera Board with M12 Mount for Raspberry Pi. Robot Shop [online]. [cit. 2019-03-16]. Available from: https://www.robotshop.com/eu/en/arducam-ov5647-camera-board-m12-mount-raspberry-pi.html," [Online].

[26] "Ultrasonic Module HC-SR04. Electronic Wings . [cit. 2019-06-04]. Available from: https://www.electronicwings.com/sensors-modules/ultrasonic-module-hc-sr04," [Online].

[27] "TurtleBot3. Robotis [online]. [cit. 2019-06-04]. Available from: http://emanual.robotis.com/docs/en/platform/turtlebot3/appendix_lds_01/," [Online].

[28] "GUDINO, Miguel. What is a Microcontroller? [online]. [cit. 2019-03-16]. Available from: https://www.arrow.com/en/research-and-events/articles/engineering-basics-what-is-a-microcontroller," [Online].

[29] "ARDUINO MEGA 2560 REV3. Arduino [online]. [cit. 2019-03-16]. Available from: https://store.arduino.cc/arduino-mega-2560-rev3," [Online].

[30] "JAIN, Arpit. Engineers Garage: Geared DC Motor [online]. [cit. 2019-03-16]. Available from: https://www.engineersgarage.com/geared-dc-motor/tags/geared-dc-motor," [Online].

[31] "Electric motor with gearbox F-GM25-370CHV-130-R. GM Electronics [online]. [cit. 2019-06-04]. Available from: https://www.gmelectronic.com/electric-motor-with-gearbox-f-gm25-370chv-130-r," [Online].

[32] "L298N Dual H Bridge Motor Driver. Spark Fun [online]. [cit. 2019-06-04]. Available from: http://www.robotpark.com/L298N-Dual-H-Bridge-Motor-Driver," [Online].

[33] "HS-311 Servo. Servo City [online]. [cit. 2019-06-04]. Available from: https://www.servocity.com/hs-311-servo," [Online].

[34] "Arduino Mega Sensor Shield. The Engineers Project: Arduino Shop . [cit. 2019-06-04]. Available from: https://arduino-shop.cz/en/arduino-platform/1202-arduino-mega-sensor-shield.html," [Online].

[35] "UART COMMUNICATION. Circuit Basics [online]. [cit. 2019-04-17]. Available from: http://www.circuitbasics.com/basics-uart-communication/," [Online].

[36] "Future of Driving. Tesla [online]. [cit. 2019-05-22]. Available from: https://www.tesla.com/autopilot," [Online].