University of Pavia
Faculty of Engineering
Department of Electrical, Computer and Biomedical Engineering

Master Degree in Computer Engineering

Design and implementation of a 3D scanner

Supervisor:
Prof. Tullio Facchinetti

Candidate:
Marko Pandža

Academic Year 2022/2023

Abstract

The swift advancement of 3 dimensions (3D) scanning technologies has brought about significant transformations in diverse sectors, such as industry, healthcare, architecture, and cultural heritage protection.

This master's thesis starts off with an introductory section that provides an overview of the topics to be discussed. This is thereafter accompanied by an extensive investigation of recent developments and their wide-ranging practical implementations. The chapter on implementation provides an extensive description of the construction of a scanner. However, it is essential for individuals to possess a theoretical background, which is further supported in the chapter on tools. The state of the art chapter presents comprehensive information about the scanner, including its design and functionality, and compares it to other scanners now available in the market.

Combining state of the art chapter with results, one comes to interesting data. The scanner gave an answer to initial question that he idea of making an affordable 3D scanner is definitely viable but one cannot expect the same results of the scanners that from 700 to 2500€. The scannable size is mostly the same and it depends on the structure of the scanners, which can always be adjusted. A big drawback of high end 3D scanner is having high requirements for the Personal Computer (PC) to run it. Main issue that was faced in the scanner is an Infrared (IR) sensor. Having an accuracy of 2 mm , comparing it to 0.1 mm of the 700€ scanner. Also a more important problem is that due to its manufacturing, it keeps scanning the object for an additional 3 to 4 mm and then not giving a proper presentation of a 3D model. By a simple change of a more expensive IR sensor, which would not have similar problems and a better accuracy, it could potentially rival its competitors.

Acknowledgments

I express profound gratitude towards my mentor, Tulio Facchinetti, for their consistent support and advice during the duration of my master's program. The individual's extensive knowledge and ability to remain composed have proven to be of immense value to me, significantly contributing to the accomplishment of this thesis.

I express my gratitude towards my friends and family for their unwavering affection and assistance during this endeavor. I owe my successful completion of this endeavor to the invaluable support and motivation provided by those who encouraged me throughout.

I would like to express my gratitude to my colleague, Valentina Ć osić , for her invaluable assistance in the creation of 3D models.

Additionally, I extend my appreciation to Ivan Aleski for their unwavering support during the entirety of the project.

Lastly, I would want to express my profound appreciation to all the individuals who took part in my research. The participants' eagerness to contribute their experiences and thoughts has proven to be of immense value to my research endeavors, ultimately contributing to the successful completion of my thesis. I express my gratitude for the allocation of your time and valuable input. I express my gratitude to all individuals who have provided me with support throughout this endeavor. This thesis would not have been feasible without the assistance and guidance provided.

Contents

List of Figures

List of Tables

Acronyms

LED Light Emitting Diode

UNIPV University of Pavia

Robolab Robotics Laboratory

V Volts

USB Universal Serial Bus

I2C Inter-integrated Circuit

IDE integrated development environment

PWM Pulse width modulation

I/O Input/Output

mm millimeters

Wifi wireless fidelity

DIY Do it yourself

UART universal asynchronous receiver / transmitter

SPI serial peripheral interface

g grams

IoT Internet of Things

BLE Bluetooth Low Energy

IR Infrared

PIR Passive infrared

ToF time-of-flight

ms milliseconds

cm centimeters

GND ground

Vcc voltage common collector

F Farad

PSD photo sensitive diode

Mhz Megahertz

MB megabyte

SRAM static random access memory

KB kilobyte

IMU Inertial Measurement Unit

DC direct current

mA milliampere

PM Permament Magnet

VR Variable Reluctance

PC Personal Computer

HPM Hybrid Permament Magnet

N North

S South

REC rare earth cobalt

DM Disc Magnet

DAC Digital to analog converter

NEMA The National Electrical Manufacturers Association

Nm Newton-metre

A ampere

FET Field-effect transistor

DMOS Double-diffused MOSFET

EMF Electromotive force

3D 3 dimensions

CAD Computer-aided Design

Nm newton-metre

m milli

Vcc Common Collector Voltage

GND Ground

ADC Analog to Digital Converter

2D 2 dimensions

Tof time-of-flight

CCD Charge-Coupled Devices

CPU Central Processing Unit

GPU Graphical rocessing Unit

RAM Random Acess Memory

CAGR Compound Annual Growth Rate

Chapter 1

Introduction

This chapter presents the idea of the thesis, combining it with the objectives. The chapter is concluded with the description of the organization of the document.

1.1 Motivation

3D scanners are widely used nowadays. The demand and use of 3D scanners is becoming more widespread each day. As a need for a 3D printer is also growing, it just means that even the need for a 3D scanner is growing daily. As it can be seen in figure 1.1 one can say the 3D scanners market will reach 1.5 billions of dollars in 2028, by having a Compound Annual Growth Rate (CAGR) of 6.9The term CAGR refers to a metric that quantifies the yearly rate of growth of an investment, while considering the impact of compounding. Within the domain of the 3D scanning market, the term CAGR denotes the mean annual growth rate of the market's magnitude or worth over a designated timeframe, typically spanning multiple years. As an illustration, in the context of the 3D scanning market, a CAGR) of 6.9% seen over a span of five years signifies that, on average, the market's magnitude or worth saw an annual growth of 6.9% throughout that specific time frame, while considering the cumulative impact of compounding. That is considered a rapid growth, specifically since it is not something that was just introduced to the public a year ago, but many years ago. It is becoming more popular as the technology is becoming better.

The scanner is needed in a situation where a high precision is needed. However, that one also requires a reliable 3D scanner.

3D scanners is also very often used in quality control and reverse engineering. Usually the more expensive the scanner is, it uses a better and more expensive

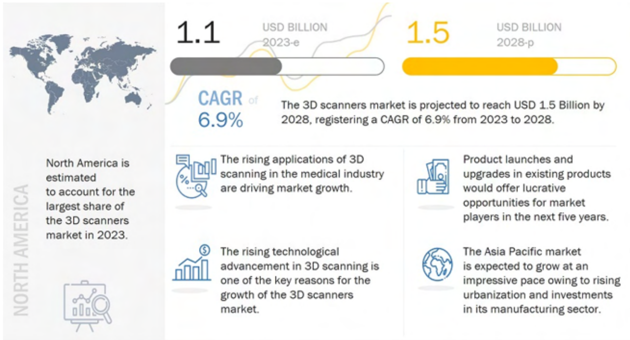

Figure 1.1: 3D scanners market forecast to 2028 [1]

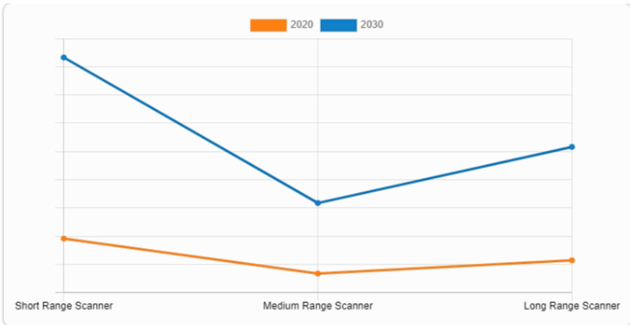

Figure 1.2: 3D scanners market forecast to 2030 by range [1]

electronics and sensors in general, which makes it better in many aspects, however a professional scanner on average costs above 20,000€. Upon looking for a 3D scanner and seeing how expensive it is, an idea came up to try make a low cost scanner to see how it can compare with other on the market. Thinking of making a short range scanner, since they seem to be the most popular ones now. However, after looking at the statistics seen in the figure 1.2 short range scanners will dominate in 2030. Thinking that they're popular now, but by the graph the popularity will 4 times until 2030. Therefore, the a short range scanner will be made in this thesis. As seen from the figure 1.2, the medium range and long range scanners together won't add up to the use of the short range scanners.

1.2 Objectives of the thesis

The objective of the thesis is to try to make 3D scanner using an Arduino platform, its software and surrounding hardware. The said technology is generally low cost, while providing a reasonable quality. So it would mean that it provides a good cost to quality ratio. Most of high end scanners require high computational power, which is usually not achievable by a normal PC. Idea is not to have an amazing accuracy and the best models that can be equal to other scanners on the market, but to be affordable to anyone, while having a minimum requirements needed for the PC, while it is also a big expense if needed to upgrade.

Considering high prices of other 3D scanners, it has been decided to try to make one and compare it to some of the known scanners on the market. Comparison will be made on the basis of price and many other parameters that are used to finding out how good a scanner is (see Section 2.2). It would be compared to an expensive one and ones that are considered to be a more affordable option. In that way, the comparison is not only made between the made 3D scanners, but also on the other scanners on the market. That way, one can see how the drop in price affects the scanners and vice versa.

1.3 Proposed solution

The scanner provided a response to the original question, indicating that the concept of developing an economically feasible 3D scanner is indeed feasible. However, it is important to note that comparable results to scanners priced between few hundreds to few thousands euros may not be achievable. The scannable dimensions remain largely consistent, dependent upon the configuration of the scanners, which can be easily modified as needed. One significant limitation of high endÂ 3D scanners is their reliance on high-performance PC systems for operation. The primary challenge encountered in the scanner is the IR sensor. Other parts are mostly mechanical, while they can have mechanical problems, those can be solved usually by a cheap solutions, but regarding the IR sensor, it does all the scanning, so depending on the one that is chosen it can make it all work perfectly or have some errors. It is quite challenging trying to make a budget 3D scanner, trying to find all the mechanical parts and sensor that will work with each other with decent accuracies. The scanner with an accuracy of a few millimiters is being compared to the scanner of a few hundreds euros, with an accuracy of a 0.1 mm. Furthermore, a significant issue arises from the manufacturing process, where not all data are well reported in the datasheet of the IR sensor, which makes it even more difficult to find the one which would fit the existing structure. However, even in the case of not choosing the perfect match of an IR sensor, one can simply substitute the existing IR sensor with a higher-cost alternative, which would not encounter comparable issues and offer improved accuracy, there is a possibility for it to compete effectively against its rivals. These scanners probably won't be used in places where a high precision is needed, such as in medicine, but it can be used in reverse engineering, in universities and highschools for an education purposes, while also bringing the technology closer to the students.

1.4 Organization of the document

The thesis is organized as follows:

- Chapter 2 compares made 3D scanner to other options on market
- Chapter 3 explains the tools used in the project
- Chapter 4 presents how a 3D scanner was made step by step
- Chapter 5 presents the results achieved by the scanner
- Finally the conclusions in Chapter 6.

Chapter 2

State of the art

This chapter provides an information about 3D scanners and compare the one made in the thesis with some well known scanners, while also giving an overview of the state of the art.

2.1 3D scanner

A digital 3D model is a numerical depiction of the object's aesthetic characteristics. A bidimensional picture of the item can be generated from the digital model in a realistic manner. Through the application of specific methods, such as perspective and shading, this image can simulate how the human eye perceives an object's three-dimensionality. A 3D visualization system typically consists of two components: the scene, which is a mathematical representation of the 3D objects, and the render, which is the method used to generate the scene's 2 dimensions (2D) pictures. The utilization of digital 3D models presents several advantages. The model can be employed for digital simulation or to generate a modified digital representation of the thing. Within the realm of entertainment, the utilization of 3D modeling enables the incorporation of tangible entities, such as items or individuals, in the process of character and environment development for digital animation. Moreover, the utilization of expandable digital models might prove to be quite advantageous in the examination of substantial entities such as structures or geographical areas.

Due to the expanding availability of 3D graphic devices and the ongoing decline in the cost of computational power, applications based on the processing of 3D models are now widely used. The digital 3D model may be derived using two distinct methods:

1. using Computer-aided Design (CAD) description

2. an actual item measurement.

The CAD method usually requires a representation of a model through various equations. It usually takes a lot of knowledge and time to get a 3D model. The method based on measurement, or the process of digitization by physical item measurement, is a method that enables the acquisition of a 3D model in a partially automated manner. The analysis relies on the quantification of geometric attributes of the item, as well as its aesthetic characteristics such as color and texture. It is generally much faster method, more accurate and generally there is no fear of a human error. In the thesis the emphasis will be placed on the second method, where an item is measured.

The second method is mostly used in following applications:

- **Reverse engineering**: It is used in many cases but if we take an example of an architecture, where a simple 3D model is much faster and cheaper to make if we were to compare it to the physical model that one would want to present to a customers [13].

- **Art**: Due to limited lifespan of an art work, a use of 3D model is desired. With the 3D model the artwork can be seen from different angles and zoomed in and out, without touching the actual art. Often used for a virtual museums [13].

- **Quality control**: The process of quantifying the properties of an item enables a systematic comparison of several objects within a shared class, hence facilitating efficient analysis and evaluation [13]. This concept can be applied to not only quality control in manufacturing but also in bio metric identification and many more.

2.1.1 Categorization

Contact and non-contact are the 2 main categories of the 3D scanners which can be seen in figure 2.1.

Contact scanners

Contact scanners employ a physical touch method for probing the surface of the item. There are primarily two types of systems used for this purpose, namely the coordinate measuring machine and the Joined arm system.

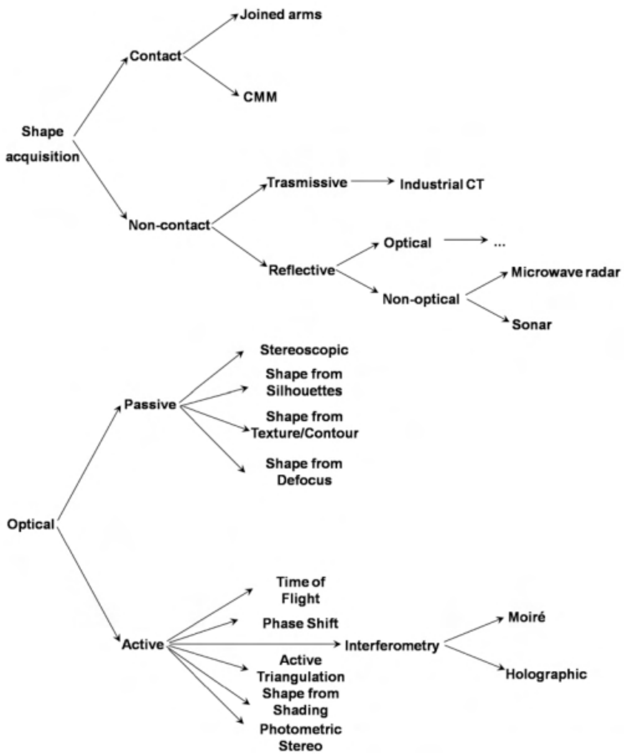

Figure 2.1: Categorization of 3d scanners

- **Coordinate measuring machine** The first consists of a tactile probe that may travel along a horizontal plane and is coupled to a vertical arm. The movement can be done in all 3 dimensions, because the systems has the 3D coordinate system, made by the 3 orthogonal axes. The object has to be placed in a specific referenced frame, before the scanning begins. While these scanners have a good accuracy, the disadvantage is that the acquisition is done only vertically and everything is bounded by the structure. It can be seen in figure2.2.

- **Joined Arm system** The Joined arm is comprised of a series of interconnected links, each possessing articulation, and is equipped with a probe as its terminal end effector. The probe's 3D coordinates are determined by the combined rototranslations performed by each connection. Having more degrees of freedom compared to the Coordinate measuring machine it can be used with a larger objects. The system is usually very sensitive to climate and is usually operated manually

While these scanners have a great precision, the big downside is that they're quite expensive and are not to be used with a fragile items.

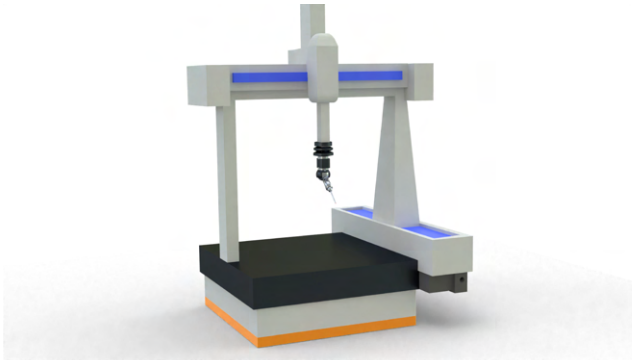

Figure 2.2: Coordinate measuring machine[2]

Non-contact scanners

Unlike contact scanners, non-contact does not need a physical touch to make a 3D model. Looking at the figure 2.1, one can see that the further division is by the scanner being either reflective or transmissive. Both of these categories use a radiation between itself and the object. The difference comes if whether the radiation should be reflected by the object and returned to the scanner or if it should pass through the object.

- **Transmissive**: The transmissive 3D scanner consist of an object, emitter and a receiver. The object is placed between the emitter and the receiver. The emitter irradiates the objects, while the receiver collects the radiation sent by the emitter but attenuated, because of the object. The primary exemplar under this classification is industrial computed tomography seen in figure 2.3. The radiation, which consists of a stream of high-energy photons produced by an X-ray tube, is able to penetrate the object of interest and is then caught by a 2D detector in the form of a digital radiograph picture. The process of reconstructing 3D models involves utilizing a collection of 2D X-ray photos of the item captured from various perspectives. The perspectives are acquired by rotating the item, which is placed on a turntable capable of spinning with a high level of accuracy, often in increments of 0.5 degrees. By utilizing a sequence of two-dimensional radiographs and employing the back-projection technique, it becomes feasible to calculate a 3D voxel model. Big advantage of these kind of scanners is that one can get a model of internal and external structure, while it does not affect the object being scanned. It is also not affected by the surface of an object being any color or being transparent, however the density of the scanner does affect this scanner. The size of the scanning is limited to 30 x 30 x 30 cm.

- **Reflective**: As already mentioned, the reflective 3D scanners take the advantage of the radiation which is reflected upon the contact with the object, which is then used to estimate the position. If it uses optical radiation (wavelength 100nm to 300μm), one considers it an optical system, while non-optical systems use other types of radiation. It can be IR, sound, electromagnetic or other. In the figure 2.1, one considers only 2 types of non-optical categories, but in fact there are many more, but these are simply the most used. These systems use a time-of-flight (Tof) principle. By knowing the velocity of the radiation which is being sent to an object and knowing that it reflects from an object and comes back. Which means that one has a double distance in this case and hence can calculate the distance. Depending on the use of the scanner one can use a sonar, radar or an IR sensor. Sonars and radars are used for depths up to 400km, while IR sensors have up to a meter. Generally sonars and radars are known for its low accuracy. Optical scanners can be divided into passive and active. Passive systems often rely on the utilization of ambient radiation that is reflected, hence avoiding the emission of any kind of radiation by the systems themselves. Typically, these methods rely on the utilization of Charge-Coupled Devices (CCD) which are conventional sensors included inside commercial digital cameras. The sensors acquire visual data of the seen environment, perhaps capturing pictures from various perspectives or utilizing diverse optical configurations. Subsequently, the photos undergo analysis to calculate the 3D coordinates of specific places within the scene. While the active systems produce a form of radiation, and the sensor captures the interaction between the item and the radiation. Based on the examination of the collected data, it is possible to determine the coordinates of the spots by considering the characteristics of the emitted radiation. Within this category, there are several types of scanners that are commonly utilized. The concepts that are most frequently employed include Tof, phase shift, and active triangulation.

2.2 Comparison

A 3D scanner is comprised of a collection of equipment and processes that have the capability to capture both the physical form and visual characteristics of an object. In general, scanners operate by employing a sampling technique to capture data

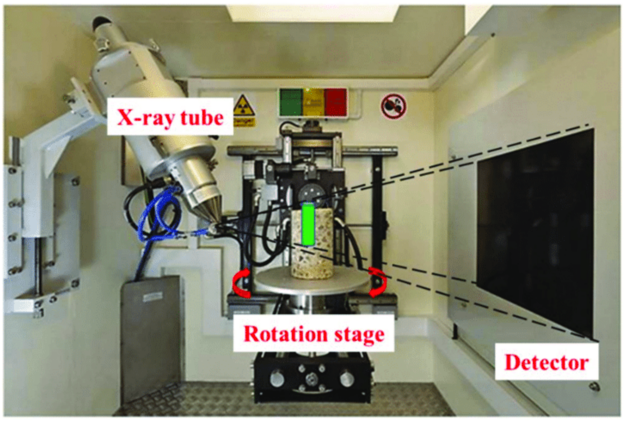

Figure 2.3: Industrial Computed Tomography[17]

from the surface of an item. The data that has been acquired is utilized throughout the reconstruction step in order to get the digital representation of the item.

Therefore, a 3D scanners can be evaluated on characteristics such as [**?**]:

- **Speed**: The evaluation of the speed of an acquisition system involves assessing the time it takes to capture a certain characteristic, such as the number of points per second that can be sampled.
- **Resolution**: The resolution quantifies the level of detail that can be seen by the system, or alternatively, it represents the minimal distance required between two characteristics in order for them to be distinguishable.
- **Accuracy**: Accuracy is a quantitative metric that characterizes the proximity of measurements to their real value. Various methodologies might be employed to ascertain the aforementioned index. There are two main classifications for measuring these entities: direct measurement, which involves utilizing a lattice with specified step increments, and indirect measurement, which is determining the average distance of sampled locations on a plane from the ideal surface.
- **Invasiveness**: The invasiveness refers to the impact that the acquisition technique exerts on the item being scanned. The act of measuring has the potential to induce various modifications on an item. For instance, incoming light can cause alterations to the surface of the object, while mechanical contact with the surface may result in damage, particularly if the object is fragile. The latter is a significant limitation for some categories of things, such as archaeological artifacts.

- **Flexibility**: The flexibility of a system refers to its ability to accommodate a diverse range of items, which may vary in many features such as materials and size. The determination of the acquisition field size and the selection of sensors utilized are determining factors.
- **Usability**: The concept of usability pertains to the level of technical proficiency required by users in order to effectively utilize a given technology.
- **Cost**: There exist acquisition systems with varying price ranges. The cost of a system is determined by the hardware and software components utilized.
- **Robustness**: The robustness of a system refers to the degree of sensitivity exhibited by the system in response to various external variables.

Just by looking at the characteristics, one can already understand that some of them cannot be measured or evaluated so they are rather a result of subjective opinion of the people who built it with some data backing its claims.

It is crucial to embark upon an investigation into the characteristics and specifications of built 3D scanner at this moment in order to establish a thorough understanding of its capabilities. It takes 653 ms for a scanner to do a 360 ° while scanning at a certain height.

$$N = steps \cdot sampling \tag{2.1}$$

Let's consider equation 2.1 which is used to calculate how samples are taken the full rotation. The motor does 200 steps and a sampling rate is set to 50 per point. By multiplying these two values, one gets $10,000$ samples taken.

$$v = \frac{N}{t} \tag{2.2}$$

To calculate the speed of the acquisition of the system one must use equation 2.2, where v stands for speed, and N for number of samples scanned over the time t. As already known the time is 653 ms, while the N is $10,000$, which means that the speed is $15,313$ samples per second. For some characteristic to be visible on the sensor the minimum difference between it and the previous one has to be 1.5 mm. For the characteristic to be see left or right close to another one, there must be a minimum 1.8 ° difference between those. However, A4988 Polulu can be set up with microstepping, which would be that even a small difference of 0.113 ° can be achieved. The said 3D scanner is not invasive, meaning that there is no contact or infiltrating of an object, which is being scanned. The downside is that it is not so

Figure 2.4: EinScan-SP V2

flexible. It can scan an object that has radius of 6 cm and a height of 15 cm. Another downside is that it cannot scan object that are translucent. It serves it purpose well, however, it is very user friendly, as one should only place the object that is to be scanned and plug the adapter into socket and the Arduino into computer. The scanner can work in most of condition anywhere inside, it should not be used outside as animals,weather may affect it. The size of the system is 180 x 120 x 240 mm .

Cost can be summarized in the table 2.1. The prices reflect the market in Croatia, and they are all retail prices. The cheapest option were not looked at, so that is a thing where the prices can be lowered down by a bit. If one can get a wholesale price, that would make an items much cheaper too.

In table 2.1, in the *Driver* item, two prices are reported. These prices reflect different drivers with similar application but different quality and significant disparity in price. For the mentioned 3D scanner Polulu A4988 was used whose prices is 17 €, however in this situation and the application that it was used for a drier A4988 RepRap - red could've been used, whose prices is only 3.9 €. It does have some limitations compared to the Polulu like max of 1 A per coil, but that wouldn't make any difference in the sense of the stepper motors that were used in making of the scanner. Also by looking at the table 2.1, one can see a category *Miscellaneous*, which consists of components such as: resistors, capacitors, Arduino wires, zip ties, nuts and screws.

Comparing the 3D scanner with a model that uses a similar principle but that is already on the market and is called *EinScan-SE V2*. It can be seen in figure 2.4

At a glance, one can tell that the principle is similar. However, this scanner

Item	Price (€)
Stepper motors	24.44
Drivers	34.00 | 7.80
IR sensor	17.50
Dampers	1.54
3D printing	5.00
Metal rods	5.00
12V adapter	6.00
Bearings	4.52
Resistors	2.00
Arduino	17.64
Breadboard	5.00
Miscellaneous	7.00
Summary	129.64 € | 102.84 €

Table 2.1: Costs of making the 3D scanner

rotates the scanner instead of the object that is being scanned, as it is done in the one made in the thesis. Table 2.2 provides full comparison.

The scan of the considered scanner is bonded by the turntable, but it also has an option to scan outside of it with limited size. It has an accuracy of 0.05 mm compared to 1.5 mm. Thesis scanner is better regarding the minimum size of an object needed to be scanned, however the EinScan is better regarding the maximum size. Scan speeds are hard to compare, cause thesis scanners uses time needed per 0.05 mm, while Einscan uses 45s per turn, which depends on the size of the device, can take longer or shorter than the thesis scanner. Generally the device size of the EinScan is bigger, while having the similar power supply. All mentioned is the objective comparisons that one can make.EinScan-SE V2 There is a big price gap between prices, comparing 102.84 € to 2299 €. Subjectively one can say that EinScan is a bit more user friendly, because and STL file is generated automatically, there's no need to use another program. Again subjectively, has a better design of construction in the sense of transportation. The inclusion of an elongated arm in the depicted figure 2.4, housing the scanner, appears to have potential drawbacks, since it could be susceptible to injury and hence impair the overall integrity of the system. Another EinScans drawback is that is has high requirements for the PC which are: Win7, 8 or 10 (64 bit) from operational systems.

Category	Thesis scanner	EinScan-SE V2
Accuracy	1.5 mm	0.05 mm
Scan mode	Turntable bound	Turntable bound and fixed scan
Minimum scan	20 x 20 x 20 mm	30 x 30 x 30 mm
Maximum scan turntable	100 x 100 x 180 mm	200 x 200 x 200 mm
Maximum scan fixed	N/A	1200 x 1200 x 1200 mm
Scan speed	653 ms per 0.05 mm	45s per one turn
Power supply	12V	12V, 3.3A, 40W
Device size	180 x 120 x 240 mm	570 x 210 x 210 mm
Weight	1.15 kg J	4.2 kg
Price	102.84 €	2299.00 €

Table 2.2: Comparison of the Scanners[1]

Regarding the hardware, it requires a Central Processing Unit (CPU) Intel Dual Core i5 or higher,while needing a minimum of 16GB Random Acess Memory (RAM)s and a Graphical rocessing Unit (GPU) Nvidia GTX660 or higher, where video memory is minimum 2GB. The stringent nature of these criteria may provide challenges for the average consumer, since they typically do not possess such specifications on their personal computers. Consequently, this may result in a reduction in the potential client base. On the other hand, the thesis scanner is friendly regarding this issue. One can use it in all 3 operational systems: Linux, MacOs and Windows. CPU Pentium 4 should be used, and basically any integrated GPU will do while having a min of 4GB of RAMs. Any PCs that is even 10 years or older should have more than enough to be able to use the scanner.

Generally speaking the EinScan scanner is better by almost all categories, however the price is 22 times higher and has a lot of requirements for the PC that needs to run it.

Looking at more affordable version of 3D scanners, the *Matter and Form 3D Scanner V2* is a good option to compare with. The full comparison is summarized in table 2.3 Both scanners are turntable bound, which means that anything that is of a proper size placed on the turntable can be scanned, but it cannot scan anything outside of the turntable. Matter and Form scanner can scan object that are larger. However, the system is also larger including the weight. Scanning speed of minimum 65s is hard to compare, since it is unknown the size of an object that is scanned in

[1] EinScan source:https://www.einscan.com/desktop-3d-scanners/

Category	Thesis scanner	Matter and Form 3D Scanner V2
Scan mode	Turntable bound	Turntable bound
Accuracy	1.5 mm	0.1 mm
Maximum scan	180 mm height and 120 mm diameter	250 mm height and 180 mm diameter
Scan speed	653 ms per 0.05 mm	minimum 65s
Power supply	12 V	100 - 240 V
Device size	180 x 120 x 240 mm	345 x 210 x 345 mm
Weight	1.15 kg J	1.71 kg
Price	102.84 €	700 €

Table 2.3: Affordable and thesis scanner comparison[2]

that time. Comparing the prices one can see that there is almost a 7 times difference in price. Requirements that are needed for a PC are similar for both scanners.

[2]Matter and Form 3D Scanner V2 source:https://matterandform.net/scanner

Chapter 3

Tools and Frameworks

This chapter presents most of the tools that were used in a project. A variety of software is used for creating and editing models, but also a software for writing a code and giving a meaning to all the hardware components. Regarding hardware, there are components ranging from a microchip that is used to program, all the way to the simple plastic which also has a crucial role in the project. Theory has to be discussed so that one can understand what is said in the rest of the thesis.

3.1 Arduino

The Arduino is an open-source electronics platform that consists of both hardware and software. The hardware includes a microcontroller board, while a software includes a software development environment. Having both hardware and software while also being open-source allows both novice and professionals to use it. The boards make connections for programming and incorporating into other circuits as easily as possible by using single or double-row pins or female headers. These could be paired up with shield-style auxiliary modules. An Inter-integrated Circuit (I2C) serial bus may allow for each one of those shields to be addressed individually as well as layered shields. Since most arduinos are powered by 5V, they have a 5V line regulator.

Arduino board can be powered via Universal Serial Bus (USB) or external supply(in most cases cannot exceed 12V. It is also programmed via USB (USB to serial adapter chips are used). It has its own software called Arduino integrated development environment (IDE),where code is written and then forwarded to the arduino via USB. The most common logic levels used in arduino boards are either 3.3V or 5V. It was developed in Italy by a group of students.

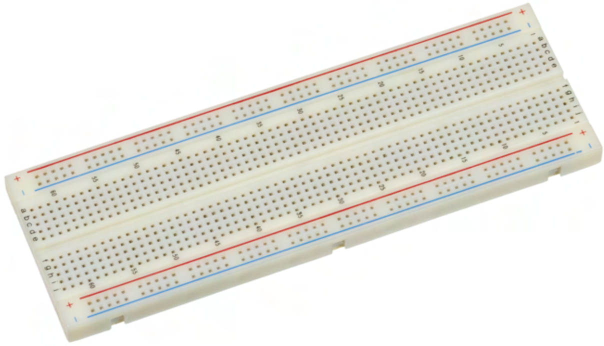

Figure 3.1: Breadboard[1]

3.1.1 Basic elements related to Arduino

Arduino has numerous components that can be used with it. However, only a some of those components are discussed as they're used in the project.

- **Breadboard**. The term describes the component used for creating and testing circuits. It resembles a little white board with several holes that are square in shape, which can be see in figure **??**. Through the breadboard's holes, wires are entered, and they are connected on the metal strips below. By using a breadboard, users may hold wires and connectors in place without having to solder them.

- **Wire**. The physical connectors that allows the movement of electricity or data are called wires. It can be seen in figure **??** as a green component.

- **Capacitor**. In a circuit connection, this little component stores and releases electrical charge. 2 charging plates and an extra component that regulates electrical discharge are often included. They can be made with a variety of characteristics and some are offered primarily for storage because of their sizes. In figure **??** the left component which has one shorter and the other longer leg is a capacitor. Farads are used to represent store charge capacity.

- **Resistor** As its name suggests, it resists the flow of the electricity and is inversely proportional with the current. In figure **??** on the right side, an component that has a small bump in the middle is a resistor.

$$I = U/R \tag{3.1}$$

[1]Source: https://en.wikipedia.org/wiki/Breadboard.

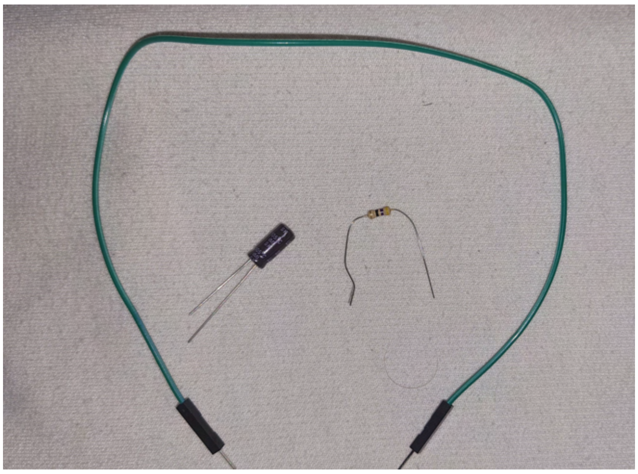

Figure 3.2: A wire,capacitor and a resistor

In equation 3.1, I stands for the current, while U is the voltage and R is the resistance.

- **Analog Pins**. They are used to transfer data or signals from an analog sensor to the computer or vice versa. Even though they're analog pins, they are eventually converted into digital data, which is then shown on the computer.

- **Digital Pins**. These pins are used for Input/Output (I/O) devices and reading the digital signals and giving a meaning to the signal to an output.

3.1.2 Features

The Arduino board has a number of characteristics that make it popular in the maker and Do it yourself (DIY) communities, including:

- **Microcontroller**: The heart of the Arduino is a microcontroller, most of them have a Atmel 8-bit microcontroller(ATmega168,Atmega328 and others), which provides the processing power,controls the input/output operations,executes written programs, handles interrupts,facilitates serial communication(universal asynchronous receiver / transmitter (UART),I2C,serial peripheral interface (SPI)) and manages the power of the board among many other things.

- **Input/Output (I/O)**: The I/Opins are needed to connect sensor, actuators or any other basic electronics into a circuit with an arduino. Therefore, a range of digital and analog input/output pins are included on Arduino boards, enabling these connections. Usually some of the pins can produce Pulse width modulation (PWM)signals

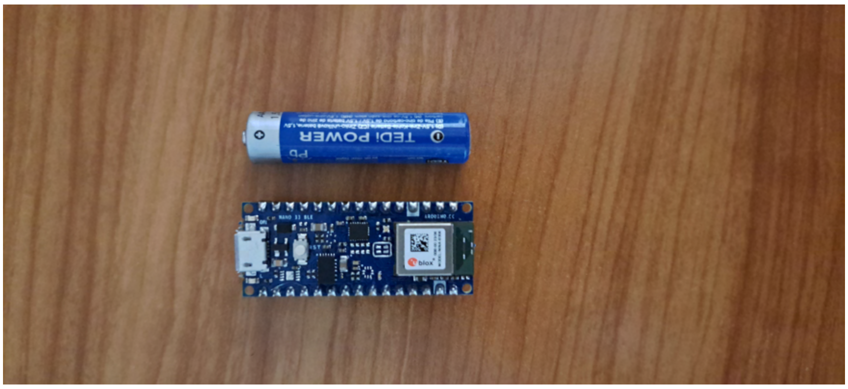

Figure 3.3: Size comparison

- **Simplified Programming**: Since the Arduino programming language is a variation of C/C++, writing code to communicate with the hardware on the board is rather simple to understand. Instead of having to worry about addresses compared to similar embedded electronics, in arduino each pin can be simply declared as an Input or an output and as a number of the pin that is currently used. There is a text console for errors, compiles in one click,uploads in one click also.

- **Extensibility**: By being so versatile it can be enhanced by external hardware modules, which provide more complexity and functionality compared to the previous version. The example are things like :Wifi, Bluetooth and motor control.

- **Open Source**: Both software and hardware of the Arduino is open source, which means that everyone can contribute to it, while (software) being free. There is a community working on fixing bugs, improving the certain design or simply enhancing it

- **Portable**: Arduino nano, which is one of the most popular arduinos has the dimension of 45mm x 18mm,while weighting around 7g without headers,making it extremely easy to carry around.

- **Affordable**: Arduinos are affordable option compared to some its competitors. An arduino can be bought already with €12 making it great and cheap investment in projects.

Figure 3.3 shows the size comparison between arduino nano 33 Bluetooth Low

Energy (BLE) and AAA[2] battery.

3.1.3 Applications

The Arduino platform is used in a variety of fields, such as:

- **Home Automation**: Arduino can be used to automate tasks in houses or apartments like lights, feeding pets and many other things [12].
- **Internet of Things (IoT)**: Arduino is a good choice for creating IoT devices and connecting them to the internet due to its connectivity possibilities and low power consumption [9].
- **Robotics**: Due to its I/O capabilities, Arduino is a well-liked platform for developing robots, handling sensors, motors, and pairing with other robotic systems [16].
- **Education**: Arduino is often utilized in educational settings in order to introduce students to programming, electronics, and prototyping thanks to its accessibility and ease of use. This project is the best example of it used in an educational purposes.

In this thesis, the arduino board and the platform are one of the key components in the setup. It allows us to gather the data from sensor and control the external devices which are further connected to some mechanical components. The cost, flexibility and ease of use was the main factor in choosing arduino as main part of the setup.

3.1.4 Arduino Nano RP2040 connect

A capable development board built on the Raspberry Pi RP2040 microcontroller is called the Arduino Nano RP2040, which can be seen in figure 3.4. It provides an efficient and adaptable framework for embedded systems and IoT. Additionally, a broad variety of I/O choices are supported, including USB, analog inputs, PWM channels, digital pins, analog pins, SPI, I2C. It works on the 3.3V logic. Connection is established via micro-USB cable. direct current (DC) current per I/O pin is 12mA.

As it has many advantages over othe microchips or even other arduinos, here are some of the most important features, why one should choose it over others. The features are:

[2]AAA is a standard for a battery of a size 10.5mm in diameter and 44.5mm in length

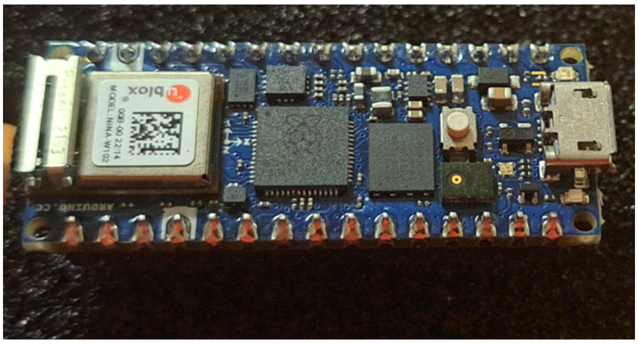

Figure 3.4: Picture of arduino Nano RP 2040 connect

- **Powerful**: With 264KB of static random access memory (SRAM) and 2MB of onboard flash memory, the board offers enough of resources for a variety of applications. The dual-core Arm Cortex-M0+ processor found within the RP2040 microcontroller operates at 133 Megahertz (Mhz).
- **Built in sensors**: It comes with Microphone,Bluetooth,Temperature sensor, Inertial Measurement Unit (IMU) and many other.
- **Raspery pi compatible**: The complete software ecosystem for the RP2040 is fully supported by it.
- **Compact**: It's small,weighs 6g but robust. Size is identical to the Arduino Nano 33 BLE,which can be seen in figure 3.3 comparing the size to the AAA battery
- **Many I/O pins**: It has 22 I/O pins. Where 20 of those are PWM, while 8 of them are analog.
- **Affordable**: The price of the arduino is 25€, which is a great option, considering all the things that come with it

3.2 IR sensor

IR sensor is a device that uses IR radiation to gather the information of its surroundings. The sensor works by emitting or receiving IR radiation or both. Usually these types of radiations are thermal and only IR sensor can pick it up, while it is not visible for humans. Two types of IR sensors can be distinguished:

- **Passive infrared (PIR) sensor**: A PIR sensor is a dual-element pyroelectric sensing device that only responds to fluctuations in heat sources. The term

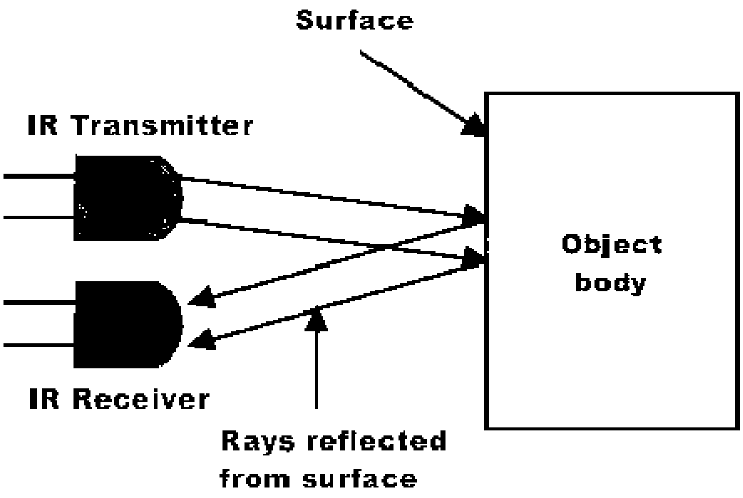

Figure 3.5: IR working principle [11]

`passive` simply means that it only receives IR radiation and doesn't emit any. An example of such fluctuations could be a movement of an animal. Animals usually have an internal temperature which can be measured and usually is higher than the temperature measured in its environment. All objects generate the IR radiation which can't be seen by humans. The electrical charge is produced ,when the sensor is exposed to IR radiation. Since PIR sensor consists of 2 pyroeletric elements, the amount of the IR radiation received by 2 of them is never the same,specially when there source of the heat is in motion. It is feasible to determine the difference in the quantity of heat received and, consequently, the motion of the heat source by utilizing a differential amplifier between the 2 components. It's to be said that PIR sensors can be divided into thermal and quantum IR sensors. The quantum IR sensor is wavelength dependent and has a fast reaction and detection time. Contrary to quantum sensor, the thermal sensor isn't wavelength dependent, but has much slower reaction and detection time compared to the quantum sensor.

- **Active IR sensor**: Active sensors are better at giving out more information than the PIR sensor. The PIR is good at detecting the motion but that's its limit. The standard active IR sensor has a transmitter and a receiver facing in the same direction. This way,when the transmitters IR radiation reflects, the receiver can it.

Figure 3.5 depicts the standard principle how the IR works.

3.2.1 Applications

Both PIR and active IR sensors have many uses in different fields, such as:

- **Motion detection**: As already mentioned numerous times, PIR sensors are great for motion detection which can be used in security purposes or anywhere where the detection of a motion is needed [15].
- **Automatic doors**: Another great usage of PIR sensors are automatic doors, where the PIR sensor can detect the individual and open or close the doors [5].
- **Distance Measurement**: Mostly active sensors like time-of-flight (ToF) are used to detect distance by calculating the time needed to receive the signal. This would be a great example of it used in a environment where it is used as a distance measurement.
- **Biometric identification**: Active IR sensors, including iris scanners and facial recognition systems, gather and analyze certain biometric traits for recognition and authorization purposes by using IR light and sensing [7].

3.2.2 IR Sharp sensor

The sensor that is used in the thesis is Sharp GP2Y0A41. By utilizing the active IR sensing principles, this sensor provides trustworthy and precise distance detecting capabilities. The active transmission and detection of infrared light is the basis for the IR Sharp GP2Y0A41 sensor's operation. It has an IR emitter built in, which directs a concentrated beam of IR light towards the intended target. The light is emitted, hits the item, and then bounces back toward the sensor. The reflected IR light is picked up by the sensor's photodetector, which then turns it into an electrical signal. The sensor determines the distance to the item by examining the signal's properties, such as strength and time delay. The sensor's analog output voltage reflects the distance being measured.

Features:

1. **Affordability**: One can buy the sensor for only €12. Comparing it to other IR which can cost double that or tripple
2. **Quick measuring cycle**: The cycle is quite fast with only 16.5ms±3.7ms
3. **Distance range**: It can measure in the range of 4cm to 30cm

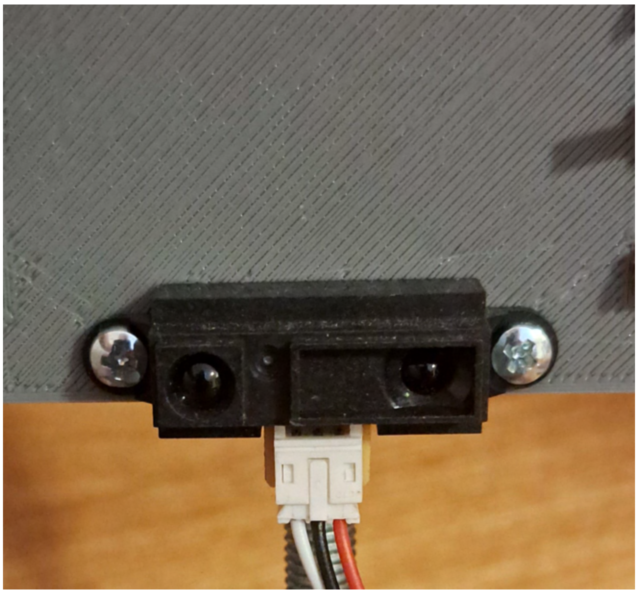

Figure 3.6: IR Sharp GP2Y0A41

4. **Portable**: With the size of 29.5mm x 13mm x 13.5mm it is quite small, can fit in the palm and can be easily carried with oneself.

5. **Analog output**: The output is given as a voltage, which is then referenced to the voltage to distance graph to acquire the actual data.

6. **Arduino compatibility**: It is compatible with an arduino software, which means that it is open to many different libraries and people who keep improving its accuracy

7. **Calibration**: The datasheet of the sensor gives a good approximation of the voltage to distance ratio with a simple equation. However if one needs less range than the sensor offers, one can calculate its own equation for a graph to get better data

Figure 3.6 shows IR Sharp GP2Y0A41 sensor firmly mounted.

Technical data: IR sensor has an operating range between 4.5V and 5.5V. It can measure from 4cm up to 30cm correctly. Theoretically it can measure also further and closer than said, but it cannot be considered truthful data. Each measurement takes 16.5ms±3.7ms. It can withstand voltages of -0.3 up to 7, as well as temperatures from -40°C to 70°C. It weights only 3.6g. A 10µF bypass capacitor should be added between Common Collector Voltage (Vcc) and Ground (GND) to make power supply more stable.

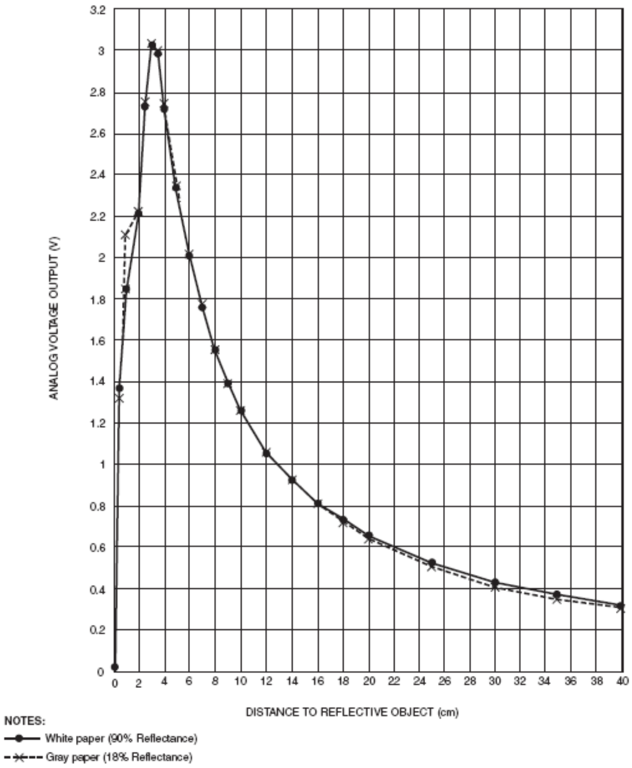

Figure 3.7: Voltage to distance graph[3]

Voltage to distance graph: Figure 3.7 displays IR Sharp GP2Y0A41 sensors graph. With the graph it can be seen that each output,which is always voltage,corresponds to a certain distance value. By looking at the graph it's clear to see why the sensor has a range of 4 to 30cm. It's because The graph can be approximated by a simple exponential function for those values. When it is approximated, any values above or below the range are not going to be correctly adressed, because the approximated graph doesn't resemble a shown graph outisde of the range. The formula 3.2 is used to approximate graph.

$$D = 13 * V^{-1} \tag{3.2}$$

Equation 3.2 is mathematical approach of approximation of the graph. Where D stands for distance, while V stands for voltage.

schematics: :

Figure 3.8 portrays a simplified block scheme diagram of the IR Sharp GP2Y0A41

[3]Source:https://www.pololu.com/product/2464.
[4]Source: https://www.pololu.com/file/0J713/GP2Y0A41SK0F.pdf.

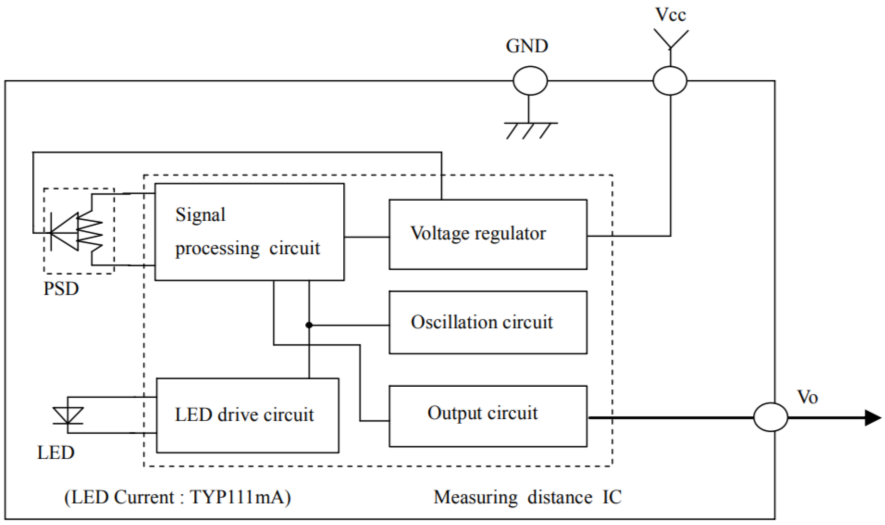

Figure 3.8: Schematics of IR Sharp GP2Y0A41[4]

sensor. Where each electronical part is shown as a block. LED shoots a signal, which is bounced back an received by a photo sensitive diode (PSD). The signal is processed and given as an voltage output to arduino.

3.3 Stepper motor

Given how unique stepper motors are, it is difficult to find a complete definition that describes it well, however stepper motors can be summed up in the following way: "A stepper motor is a brushless DC motor whose rotor rotates in discrete angular increments when its stator windings are energised in a programmed manner. Rotation occurs because of magnetic interaction between rotor poles and poles of the sequentially energised stator windings. The rotor has no eletrical windings, but has sailent and/or magnetised poles" [3]. Following up from the definition it can be said that the stepper motors are digital actuators well suited for the use in digital control systems. Stepper motor has a stator winding that is designed to be energized as its input and a discrete angular rotation as its output. When the current in one or more of the stator windings is switched, a stepper motor executes the task of moving the rotor via a precise angular step,therefore the name stepper motor.

3.3.1 Operation of a Stepper motor

Stepper motors are generally divided into 2 types: with and without Permament Magnet (PM). Figure 3.9 shows the further division of the stepper motors after the

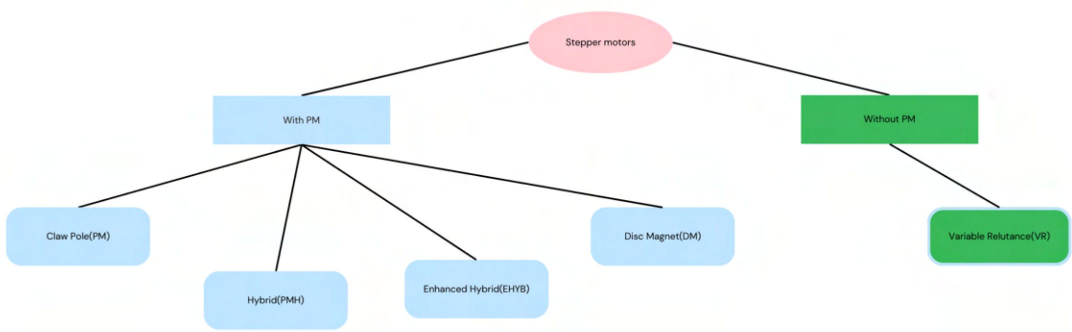

Figure 3.9: Types of Stepper motors

use of PM. Each type of motor will be discussed.

Variable Reluctance (VR). As it can be seen from the figure 3.9, VR stepper motor is a type that has no PM. The beneficial characteristic of VR stepper motors is that the magnetic alignment force is unaffected by changes of the field direction.It is so because, the torque depends on the magnitude of the current rather than its direction. As a result, a phase's current just has to be turned on and off; it is not required to be reversed. The cross section in figure 3.10 has a rotor with 4 teeth(2 Xs and 2 Ys) and a stator with its teeth. These stator teeth extend all the way to the rotor airgap. Each tooth has winding that when it is stimulated by a DC current, emits a radial magnetic field. In order to create 1 phase, the windings on the opposing teeth are joined. By having 6 stator teeth, it means that there are 3 phases, which is also the minimum number of phases required to perform a rotation in any direction. The primary flux path for a single phase excited device is from 1 stator tooth, over an airgap into a rotor tooth, directly across the rotor to another rotor tooth-airgap-stator tooth combination, and back to the stator tooth via the back of the stator. Although it is unlikely, a little percentage of the flux might 'leak' through idle stator teeth. The scenario in figure 3.10 illustrates current running through winding 1 on stator, since it is energised the rotor teeth X are attracted to windings poles. If one wants to make a 30 degrees clockwise, it can be done by running a

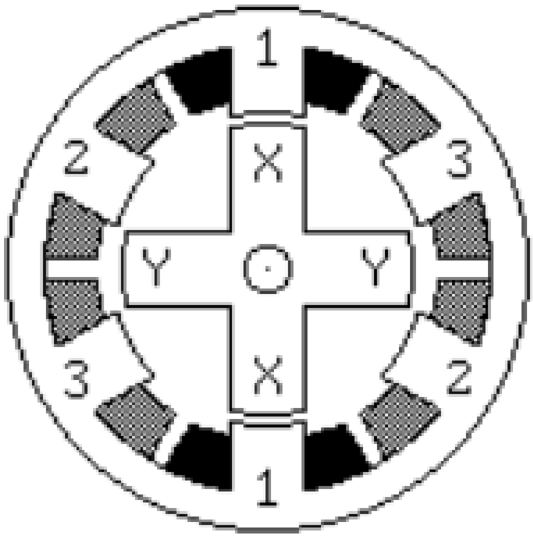

Figure 3.10: cross section of a VR motor[10]

current through winding 2. By doing so, the rotor will move so that winding poles marked with 2 line up with rotor teeth Y. This motor requires sequential application of electricity to the 3 windings in order to revolve continuously. When the rotor changes a position due to excitation of a new pair of windings, it is done in a way which minimizes the main flux path reluctance. In equation 3.3, N is number of steps,while p stands for number of rotor teeth.

$$steplength = (360/Np)^\circ \tag{3.3}$$

By applying the equation 3.3 to figure 3.10, one gets a step length of 15 degrees. Stator teeth require be an even multiple of the number of phases

Claw Poles. It's gotten its name,because the steel plate of the motor resembles the claw. The windings for claw pole motors are made out of plain, circular coils wrapped on plastic bobbins. Soft iron bits punched together make up the stator. The magnetic flux is directed by claws to the proper regions within the air gap. They construct the stator poles. These claws are either north pole or south pole, depending on the flow of current through the windings. The inductor is magnetized to the North (N)-pole firstly,following by the South (S)-pole, then to the N-pole, and finally to the S-pole when the coil is powered. A typical PM motor of the claw-pole type contains 24 inductors (claw poles) in phases A and B. For the coil, bi-filer

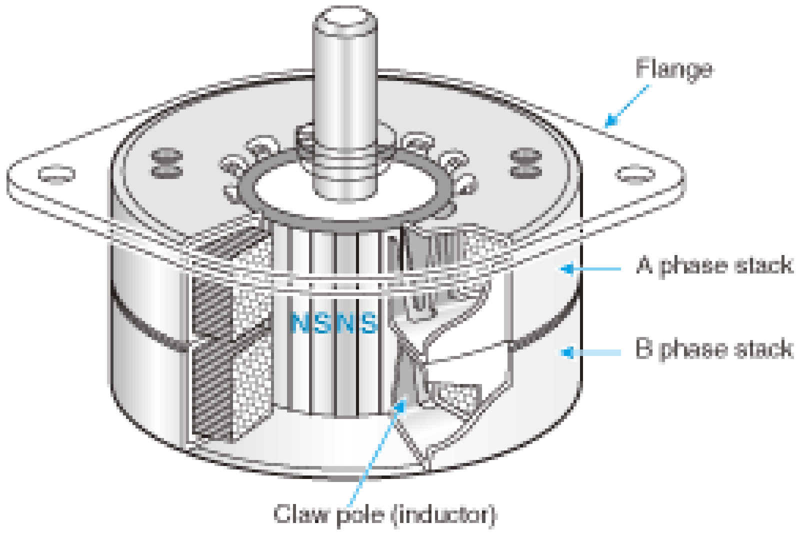

Figure 3.11: Structure of claw pole[5]

winding is often used. It is known for its low manufacturing cost and that's why it is used in home applications including PCs too.

HPM. HPM is the most used stepper motor. In this case, one can take a look at an example in figure 3.12 of a stepper motor with 4 stator poles, while each pole has a winding. Phase A winding is created by having windings on poles 1 and 3 linking together in a series. The same approach is used to form a phase B with windings 2 and 4. There is a magnet on the rotor having HPM and HPM poles.

In figure 3.12(a) one can assume a voltage +V being applied to the A- phase winding due to this, stator pole 1 will be N and pole 2 will be S. As observed in the figure 3.12(a), this results in a stator magnetic field vector F_a. The rotor must reposition itself, so that there is a lock to its N-pole to the S-pole of the stator and rotors S-pole to acN-pole of stator. Poles 2 and 4 will become N and S poles, respectively, when a voltage of +V is added to winding B while winding A is still energized as before. As seen in figure 3.12(b), this results in the creation of another stator magnetic field vector F_b. When this occurs, the stator magnetic field vector F will be at an angle of +45 degrees to where it was before. As a result, the rotor will rotate over a set 45 degree angle. Now one can take a case where winding B is still as energised as before, but winding A is deenergised. With that F_a is equal to zero and the resulting vector being equal only to F_b. As seen in figure 3.12(c) the rotor needs reposition itself by another 45 degrees to align itself with a vector. In figure 3.12(d)

[5]Source: https://www.nidec.com/en/technology/motor/basic/00031/.

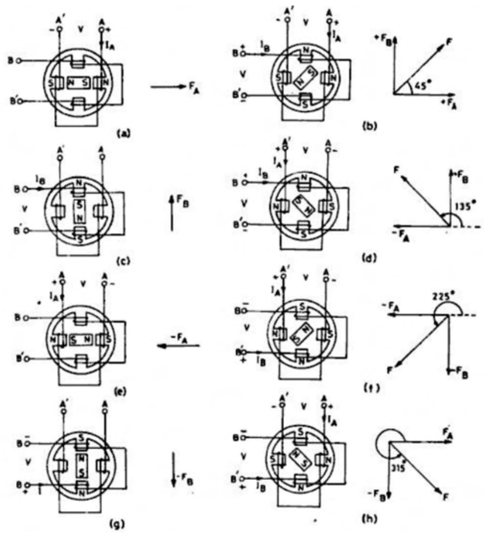

Figure 3.12: Operation of HPM[6]

one can assume a case where winding B is again energised as before, but now on phase A a voltage -V is applied. As a result a new vector F_a is generated which has the opposite direction of the one in figure a when it had +V applied to the phase A. Rotor again has to reposition itself by 45 degrees to align itself with a vector. The similar principle applies to other figures, where there is switching between energising with +V or -V or denergising and as a result making the rotor make a step of 45 degrees and eventually making a full circle.

HPM. It is simillar to the HPM, but there is usually NdFeB PM pieces inside of stator or some kind of rare earth cobalt (REC). Due to this change the airgap flux is crossed directly from teeth of the rotor to the stator teeth, which means that airgap flux can't leak. Enhanced HPM also have a lower power consumption while having higher static torque.

Disc Magnet (DM). Simillary to the enhanced HPM, the rotor of the DM is made out of a REC. Numerous changing N and S poles are imprinted by a selective magnetization procedure.

Looking at the figure 3.13 one can assume that at the beginning both phases A and B are off. in figure 3.13(a) phase A is connected to the +V, while the B phase is still off. As seen in figure 3.13, this results in the N and Spoles, which also causes the rotor to rotate and reposition itself by half a pole in order to lock its poles with the opposing poles of phase A (N-S, N-S). figure 3.13(b) In order to produce the N and S poles of the stator, phase A is switched off and -V is applied to the B phase.To

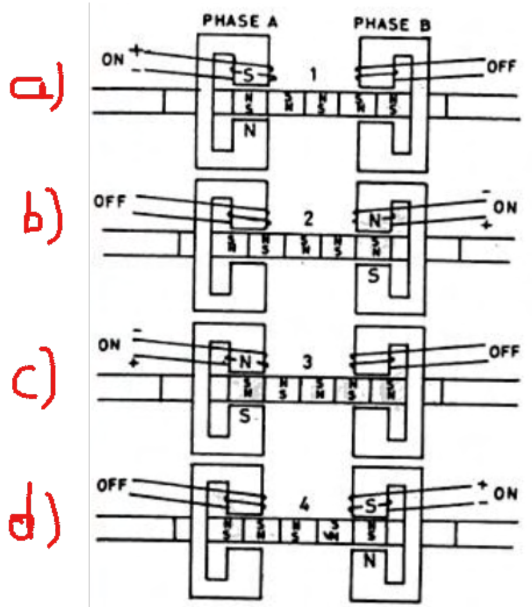

Figure 3.13: Operation of DM[6]

lock its poles to the opposing poles of the stator, the rotor must reposition itself by a half a pole. In the third step in the figure 3.13(c) , the rotor moves once again to the right by half a pole as -V is linked to A phase while B phase is disconnected. In the figure figure 3.13(d) the rotor is further moved to the right by half a pole in the last stage as a result of de-energizing phase A and connecting phase B to +V. Phase windings A and B should be positioned at the tips of the stator poles, with the rotor magnets facing straight into the airgap, to produce maximum flux in the airgap and the maximum torque.

Electrohydraulic. In a closed loop system, an HPM or VR motor is coupled to a hydraulic servo motor to create an electrohydraulic stepper motor. A pilot valve is rotated by the electric stepper motor each time it completes a step. The valve facilitates high pressure fluid to enter a hydraulic motor's cylinder. The spool travels in a way that closes the pilot valve when the hydraulic motor turns. This stepper motor can generate very high torques while operating at high stepping rates because of the hydraulic motor it uses.

Unipolar vs Bipolar. It is another division between stepper motors, which is important to notice in the thesis.

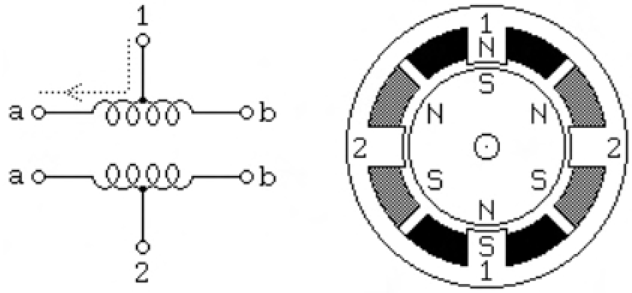

Figure 3.14: Structure of Unipolar motors[10]

Unipolar. Unipolar motors are usually wired as seen in the figure 3.14 with a center tap on each of the windings. The 2 ends of the windings are alternatively grounded to invert the direction of the field produced by that winding, while the center taps are traditionally wired to the positive supply. The step angle is 30 degrees in the motor from the figure 3.14. There are 2 windings, where the first 1 is placed between the top and bottom stator pole, while the second one is placed between left and right stator poles. There are 6 rotor poles. Top stator pole is N and bottom is S, because there is current flowing from the center tap of the first winding. For this reason rotor is in the position it is showed in the figure 3.14. When the positive supply is disconnected from the first winding and applied to the second, the rotor has to reposition itself by 30 degrees to have a N-S poles attraction between stator and rotor poles. The next step would be to apply the positive voltage to the bottom of the first winding, while disconnecting the previous winding. This results in rotor making a 30 degree step again. The fourth step is to disconnect the previous winding and apply a voltage to the second winding to the left. In this sequence it can be seen that only 1 winding can be energised at the moment of time. It is possible to have a different sequence where 2 of them can be energised at a time, but it doubles the power consumption while increasing the torque by 1.4 times.

Bipolar. Bipolar motors are almost the same as the unipolar, but the difference is in how the windings are wired, hhich can be seen in the figure 3.15. There is no center tap, making the motor easier to drive. The drive circuitry, which must switch the polarity of each pair of motor poles, is the more challenging component.

In bipolar motors, the operational sequence is identical to the unipolar, the only difference is that there is no disconnecting of the other windings, but they rather have a negative voltage given to them. One would need an H-bridge to control the positive and negative voltage. H-bridge driver chips feature control inputs for both

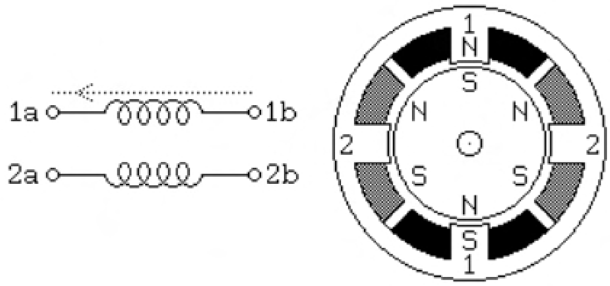

Figure 3.15: Structure of Bipolar motors

direction and for enabling the output. Usually 1 of those chips is used per winding.

3.3.2 Characteristics of Stepper motors

There are a few features that stepper motors are distinguished by. Depending on what kind of work the stepper motor should do, one would look at some of these characteristics to decide on which one should suffice the requirements.

Step angle. Step is an exact angle movement that stepper motor makes each time a different energisation happens. Number of steps for HPM are given by the equation 3.4

$$Z = 360/\theta_s \tag{3.4}$$

$$\theta_s = 360°/(N_r * K_s) \tag{3.5}$$

where N_r stands for number of teeth on the rotor, while K_s is 4 for 4 step sequence of energisation of the windings of a stator or 8 for 8 step sequence. Z stands for revolution. From the equations 3.4 3.5 one can construct a table for certain values.

The accuracy of step depends on the materials used and how it was built, but mostly it goes from $\pm 5\% - \pm 1\%$

Dynamic characteristics. Dynamics characteristics are associated with motors in a motion.

The figure 3.16 depicts a graph showing the dynamic characteristics of a stepper motor. There are multiple things to notice in the graph and new terms that are going to be explained.

N_r	θ_s	Z
6	15	24
9	10	36
12	7.5	48
15	6	60
25	3.6	100
30	3.0	120
50	1.8	200

Table 3.1: θ_s for K_s being equal to 4 for HPM stepper motors

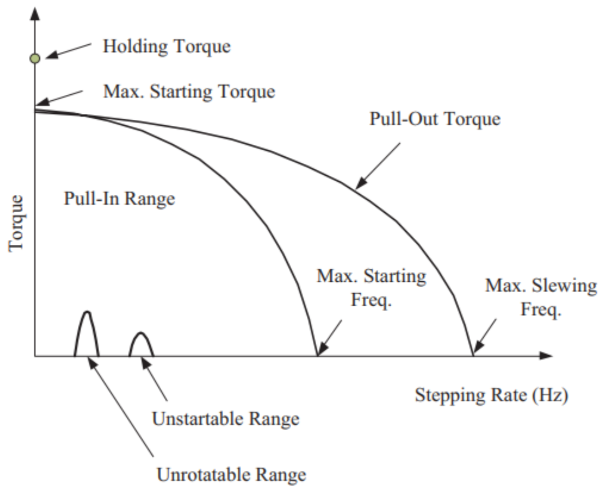

Figure 3.16: Dynamic characteristics of stepper motors[4]

- **Slewing range**: It is a range of stepping rates at which motor can run with a load torque without losing synchronism[6]

- **Pull-In curve**: This curve relates to a motion of a motor called start-stop, where a rotor comes rest after making one step. Motor can react to each individual pulse but it also stops immediately as no more energising is being done.

- **Pull-Out Curve**: It's called slewing mode of a motor and in it, the rotor still moves as a result of the previous pulse even when the next pulse already came. Contrary to the start-stop motion, slewing rate can achieve must faster rates and it can not stop immediately after the energisation is done, there will be a few more steps done before it stops.

- **Maximum Pull-Out Rate**: This is the largest stepping rate that an unloaded motor can operate at without losing steps.

- **Maximum Pull-In Torque**: It is characterized as the highest possible frictional load torque at which the motor can synchronize and start at a pulse train of very low frequencies.

- **Maximum Starting Frequency**: It seems self explanatory already, as it is the maximum control frequency at which the motor without the load can either start or stop but that it loses no steps.

- **The Starting Characteristics**: It describes the frictional load torque range for various pulse train frequencies where the motor can start and stop without losing synchronism. Motors cannot maintain a desired rotation at small frictional loads in some frequencies.

- **The Slewing Characteristic**: When a certain driver initiates a specific excitation mode in the self-starting range, the pulse frequency steadily rises and the motor finally loses synchronism. The slewing characteristic, also known as the pull-out torque characteristic, is the relationship between the torque of the frictional load and the highest pulse frequency at which the motor can synchronize. It is greatly affected by the driver used.

[6]This term means that for each pulse sent to the motor the motor makes the movement, no pulses are lost

Parameter	Value
Step angle	1.8°
Number of steps	200
Rated Voltage	12V
Rated Current	0.4A
Torque	0.25Nm
Temperature Rise	80°max
Shaft diameter	5mm
Weight	220 g
Coil resistance	30Ω
Dimensions	42 x 42 x 34 mm

Table 3.2: NEMA 17 Specifications [8]

Microstepping. Stepper motors utilize the microstepping approach to produce finer and more accurate positioning than the standard full-step or half-step modes. It enables intermediate steps between the discrete angular steps that stepper motors generally execute. Microstepping does two things, the first benefit is that it enables stepping motors to halt and retain positions halfway between full and half steps. Additionally, it substantially decreases noise at intermediate speeds, the jerky nature of low speed stepping motor operation, and resonance issues. Therefore there are less vibrations. Microstepping is usually done with motor drivers, but even then the precision is not great. The precision is affected by quantization error of the current in a winding through Digital to analog converter (DAC).

Bipolar stepper motor with 1.8° degree step. The motor of choice for the thesis is stepper motor JK42HS34-0404. It is a HPM and work with a working principle described in 3.3.1. It is a NEMA17[7] stepper motor. It has 4 leads, which indicates that it is also a 2 phase stepper motor.

The motor is compatible with arduino, has a good accuracy for full and half steps, while allowing microstepping if it would be needed. It is also a bipolar, which means that it generally has higher torque, better performance at high speeds and is

[7]It is a type of a standard of stepper motors depending on dimensions, shaft size and etc.

[8]Inspiration taken from: `https://botland.store/stepper-motors/3607-stepper-motor-jk42hs34-0404-200-stepspr-12v04a025nm-5904422350345.html`.

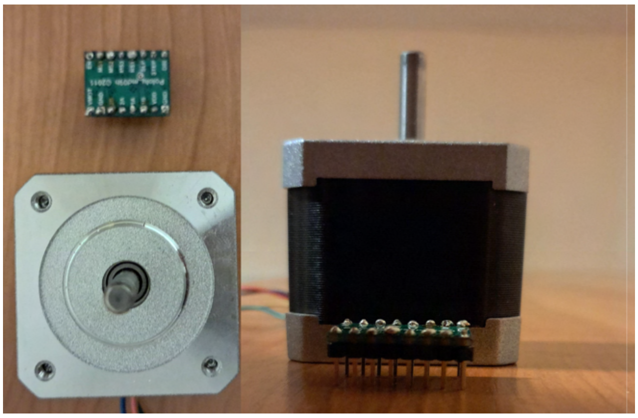

Figure 3.17: A4988 Polulu size vs NEMA 17

more efficient in the terms of losses.

3.4 Stepper Motor Driver

These devices are used to provide better control of the motors. It's a digital device and therefore, can be controlled via softwares including ardunio IDE. The most popular one of those drivers is called A4988 Polulu[9]. It is quite small which can be seen in figure 3.17.

Operation of A4988 Polulu. The A4988 is an integrated translator and full featured microstepping motor driver that allows for simple operation with few control lines. Bipolar stepper motors may be used with it in full, half, 1/4, 1/8, and 1/16 step modes. Circuitry with fixed off-time PWM control limits the currents in all of the N-channel Double-diffused MOSFET (DMOS) Field-effect transistor (FET)s and both of the output full-bridges. The value of each full-bridge's external current-sensing resistor, a reference voltage, and the output voltage of its DAC (which is in turn controlled by the output of the translator) are are all factored in for calculating the current at each step. The translator sets the DAC and the phase polarity to the initial home state at power-up or after a reset, and it also sets the current regulator to mixed decay mode for both phases. The translator automatically moves the DACs to the next level and current polarity whenever a step command signal appears on the step input. The decay mode for the active full-bridge is set to mixed while the motor is stepping and the DACs new output levels are lower than their prior output levels. The decay mode for the active full-bridge is set to slow if the new output

[9] A4988 stands for a name of the device, while Polulu is a brand

MS1	MS2	MS3	Resolution
L	L	L	Full step
H	L	L	Half step
L	H	L	Quarter step
H	H	L	Eight step
H	H	H	Sixteenth step

Table 3.3: A4988 Polulu Microstepping Configurations

levels of the DACs are greater or equal to their prior values. By minimizing current waveform distortion brought on by the motors back Electromotive force (EMF), this automatic current decay selection enhances microstepping performance.

STEP [10] It is a low to high transition signal. After the step input signal the motor makes an increment. The increment depends on how the hardware is wired, so it can be from a full step all the way to a 1/16 of a step.

Direction Input. This pin controls the motor's direction, although modifications don't take effect until the following STEP.

Reset Input. It disables all of the FET outputs and returns the translator to a predefined home state. Until reset signal is set to high all step signals are ignored in the meanwhile.

Microstepping. MS1,MS2 and MS3 pins are used for microstepping. Depending on which kind of a step is needed the certain configuration needs to be applied.

Current limiting. One must be careful with limiting the current. It needs to be adjusted depending on the motor that is used and its rated specifications.

$$I_{max} = V_{ref}/8 * R_{cs} \tag{3.6}$$

Equation 3.6 shows how to calculate the current flowing. V_{ref} is measured on the driver. $R_c s$ is a constant.

[10] STEP written with capital stands for step input signal.

3.5 Meshmixer

3D modeling and digital manufacturing have attracted a lot of interest lately in a variety of industries, including art, design, engineering, and medical. The need for capable yet user-friendly software for building and modifying 3D models has significantly increased Autodesk created Meshmixer, a powerful 3D modeling and mesh processing tool, in answer to this need. Free freeware Meshmixer offers a wealth of tools and features, making it a crucial resource for 3D design and manufacturing projects.

Mesh Editing and Sculpting. Meshmixer's robust mesh editing and sculpting features are its heart. Users may execute complex transformations and simply load 3D models in a variety of formats. Vertices, edges, and faces may be easily selected and moved about by the program. The sculpting capabilities also provide designers and artists the ability to make organic and creative changes, providing a degree of detail that is hard to attain with conventional CAD software. With its remeshing feature, Meshmixer improves mesh editing even further. Users may optimize mesh topology with the use of automatic remeshing techniques, producing geometry that is more consistent and clean. When working with complicated models or scanned data that has a mesh that is not straight, remashing is very helpful.

3D Printing Tools and Optimization. meshmixer has a number of tools designed specifically for 3D printing preparation and optimization, which is one of its key benefits. Software that accelerates the printing process is essential as 3D printing becomes increasingly common in production and prototyping. Users may examine and spot possible mesh problems including non-manifold edges, intersecting faces, and concerns with wall thickness using the analysis tools in meshmixer. By doing this analysis, it is possible to make sure that the 3D model is error-free and suitable for printing. meshmixer is also excellent at creating support. Custom supports may be created by users for overhangs and intricate geometries, which are essential for effective 3D printing. Having the ability to design optimum supports decreases material waste and improves print quality overall. The meshmixer tool also has a special hollowing capability. Users may dramatically minimize the quantity of material needed for printing, resulting in lighter products and shorter print times, by hollowing out models. The hollowed-out model's structural integrity is intelligently preserved by the program. Additionally, meshmixer has orientation optimization features that aid in placing the model in the best possible position for printing. The use of less support material, better surface quality, and less chance of warping during printing are all benefits of proper orientation.

Mesh Analysis and Utilities. Meshmixer offers a variety of mesh analysis and utility features in addition to its editing and 3D printing capabilities. For accurate design and analytical activities, users may measure 3D models' distances, angles, and volumes. Users can enhance the surface quality of the mesh by using surface smoothing techniques in meshmixer to remove defects or jagged edges that could result from scanning or mesh editing. The stitching tool in meshmixer enables the smooth mixing of several mesh components, making it easier to create intricate and linked 3D structures. This tool is useful for tasks that call for the fusion of several parts into a single cohesive product.

Chapter 4

Implementation

"Good judgment comes from experience, and
experience comes from bad judgment."

Fred Brooks

I could not have found a better quote that can be related to the hardships of the thesis. Starting from only theoretical knowledge and zero practical experience, it has been really challenging, inspirational, educational and frustrated at the same time. Looking back now at one of the first days, when I didn't know how to connect a stepper motor, one can only see how much I've learned throughout. Using different types of electronics, tools, finding a problem in much quicker time, while also immediately having an idea where the problem could lie.

4.1 Structure 3d models

The structure, where all the components are going to be placed, is very important. It must hold everything together and made precisely and measured to fit the sensor and motors. Many models were made and edited to fit the needs, but ultimately these were decided upon after many modifications. Figure 4.1 depicts the base body of the strucutre.

The stepper motors will be placed in the corners, where a hole was made specifically for its dimensions. The one that will fit a X and Y-axis motor is made a bit higher, because that is where the turntable will be and it should be higher than the IR sensor. There is a small path, which connects 2 motors, where motor wires will be placed. There is also another model similar to this, where most of the walls are cut off to make it easier to put and take out motors.

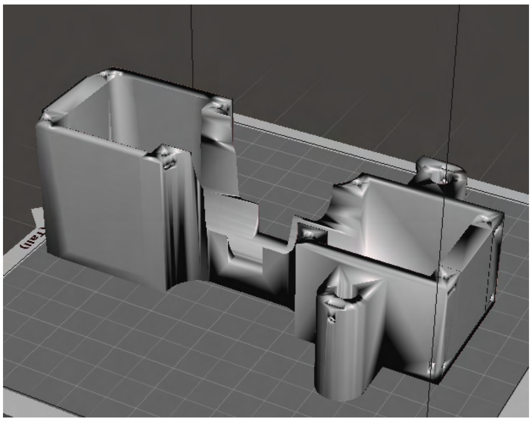

Figure 4.1: 3D model of base body

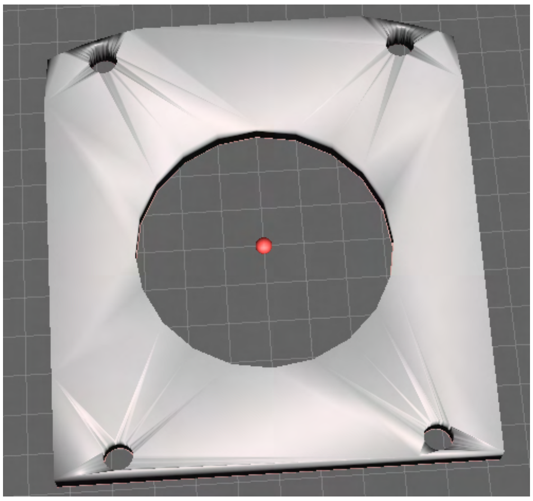

Figure 4.2: 3D model of a cover of the X and Y-axis stepper motor

Figure 4.2 shows a cover of X and Y-axis stepper motor, while figure 4.3 shows a cover of the Z-axis motor. It's important to notice, that below the cover there is a damper connected to the motor.

A coupler can be seen in figure 4.5. Its purpose is to make connection between a Z-axis motor shaft and a lead screw. On a coupler a different threads can be applied. One comes from the stepper motor which transmit the movement and on the second end of the coupler a threaded metal rod will be placed. When the motor makes a step, that movement is transmited to the threaded rod, which makes the the IR sensor go upwards.

A cover for the lead screw must be made in order for it to do the specified work that will be discussed in the section 4.4. The cover can be seen in figure 4.4

IR sensor will be placed on a holder that was made specifically for IR SHARP

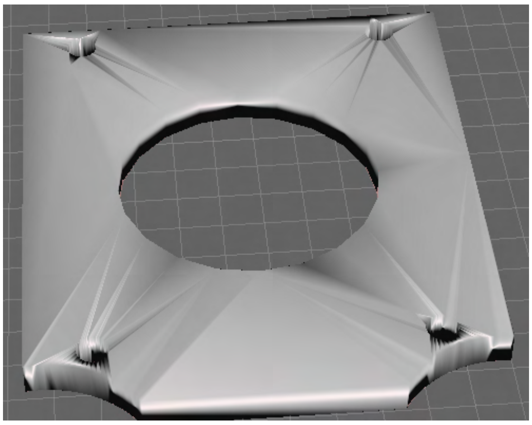

Figure 4.3: 3D model of a cover of the Z-axis stepper motor

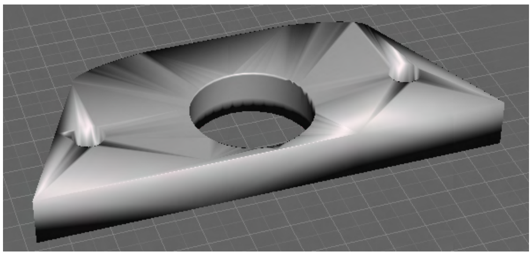

Figure 4.4: 3D model of screw cover

Figure 4.5: 3D model of a coupler

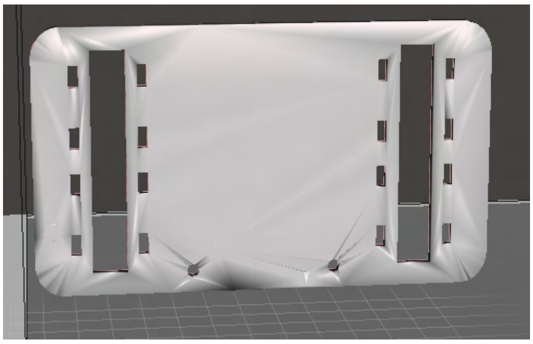

Figure 4.6: 3D model of IR sensor holder front part

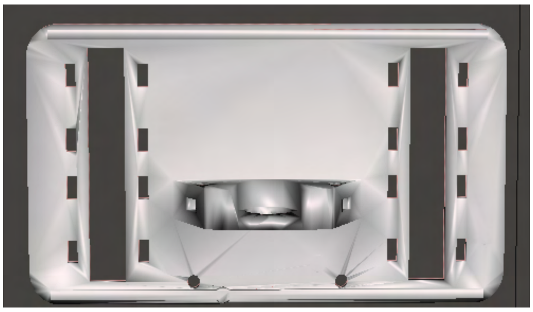

Figure 4.7: 3D model of IR sensor holder back part

GP2Y0A41, whose front and back sides can be seen in figure 4.6and in 4.7. The backside of the IR sensor holder has a hole, where lead screw will be.

A turntable is connected to the coupler which is connected to the shaft of the stepper motor in order to transmit the rotation. A rotation makes the turntable spin clockwise or anti-clockwise depending on the motor which in reality is responsible for X and Y-axis rotation. It can be seen on figure 4.8.

4.2 Stepper motors and drivers setup

After the model of the structure is done, it is time to start to work with stepper motors. The one that was used in this project is stepper motor JK42HS34-0404 200 steps 12V/0,4A/0,25Nm. The first thing to do is to find out which 2 wires make up a coil. There are 2 coils, which means that there are 4 wires. The fastest way is to look at motors datasheet, however, sometimes the datasheet provides wrong colors of wires, for example, saying that red and blue is a coil when that certain motor that is being used doesn't have a red wire. Figure 4.9 shows how to setup the wire in the correct way. By looking at the diagram red and blue wires make up a coil, while black and green make up the other one. Knowing that a driver a4988 polulu is used,

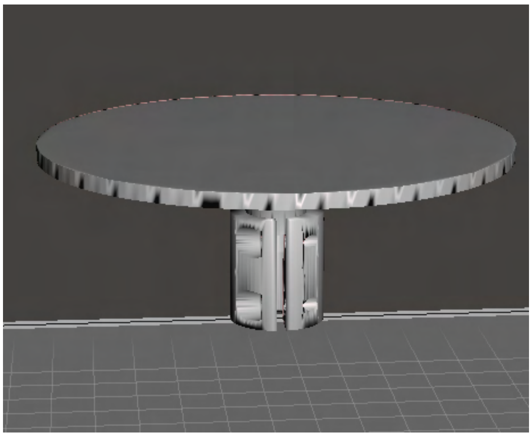

Figure 4.8: 3D model of a coupler and a turntable

it would mean that 1A 1B is one of those coils while 2A and 2B is the other pair of the wires. Due to the fact that sometimes the datasheet provides false information, 2 more ways to find out which wires make up a coil will be discussed:

1. **Multimeter technique**: First, one must se a multimeter to a *diode test* and a multimeter should display 1. Secondally, pick 2 wires and touch it with the multimeter probes (one probe per wire). Once a right pair was found, the multimeter will display a small number that is different than initial 1,which was displayed.

2. **A technique with no tools needed**: A second technique focuses on the fact that more resistance is applied when a right pair is found. One must take a shaft and rotate it to get a feeling on how much force is needed to be applied to it to make it rotate. Secondally, taking 2 wires and interconnecting them, if those are the right pair of wires it should be significantly harder to rotate the shaft.

After knowing the right pair of wires one must take notice that it will be used with Arduino. Since Arduino is usually connected to the breadboard, one must use a specially designed wires that are used for it. Therefore, one must solder the specified wire on the ends of the motor wires. Adding insulation is a must to protect it. Figure 4.10 displays a stepper motor wires soldered into Arduino wires and coated with an insulation.

After knowing the pairs of wires and soldering Arduino wires onto it, one must solder header pins of the driver onto it.

[1] Image taken from the datasheet of the motor

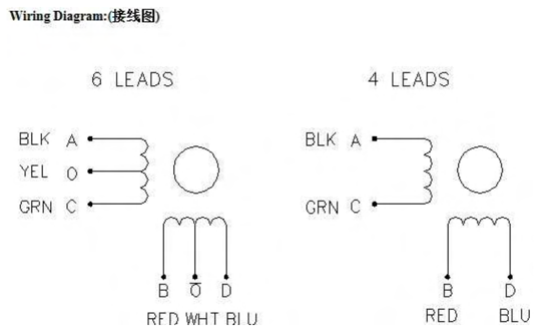

Figure 4.9: Wiring Diagram of stepper motor[1]

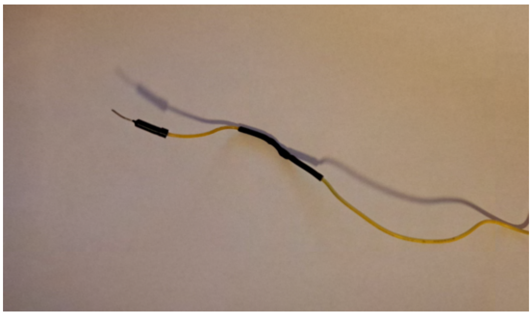

Figure 4.10: Stepper motor wire soldered

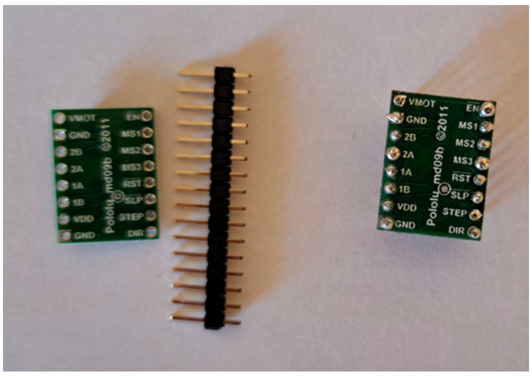

Figure 4.11: Driver with headers

Figure 4.11 shows the before and after soldering the headers of the driver.

4.2.1 Current limiting

Before connecting the stepper motor to the driver and testing it, one must never forget to limit the current of the driver according to the stepper motor used. Utilizing the highest voltage that is practical for your application is one technique to improve stepper motor performance. Since the current in the coils can change more quickly after each step, increasing the voltage typically makes it possible to make higher step rates and stepping torque. However, the coil current should be actively restricted to prevent it from going beyond the motor's rated current in order to use voltages over the stepper motor's rated voltage safely. There is a potentiometer on the driver, which is used to limit the current. Now, there are 2 ways of setting the current limit:

1. **Setting current**: One must set a driver into a full-step mode(it's done so by default) and use a multimeter to measure the current between a pair of wires, which have been previously found. This means that the current which is measured is actually only 70% of the actual current limit. A screwdriver is used to change the current running through a coil by changing the potentionmeter very carefully, because it is very sensitive. If one considers a motor that is being used in this case, which has 200 steps 12V/0,4A/0,25Nm.

$$I = 0.4 \cdot 0.7 \tag{4.1}$$

$$I = 0.28A \tag{4.2}$$

   In equations 4.1 and 4.2 one can see that I is a current that A needs to be displayed on the multimeter, when the measuring is done on a coil in the full-step mode. For microstepping mode a current through a coil is approximately larger by 42% compared to the full step mode.

$$I = 0.28 \cdot (1/0.7) \tag{4.3}$$

$$I = 0.4A \tag{4.4}$$

   Looking at the equations 4.3 and 4.4 it can be deduced that the current in the coil in the microstepping mode is equal to the rated current. Both modes were used in troubleshooting a certain errors in the project.

2. **Setting voltage**: Second way to limit the current is by adjusting the voltage and by changing the voltage, the current is also being changed. A desired

voltage must be calculated and then applied to the driver. Identically to the previous method, one must use a screwdriver to rotate the potentiometer to change the value of the voltage.

$$I_{rated} = \frac{V_{ref}}{8 \cdot R_{cs}} \tag{4.5}$$ [2]

Where V_{ref} stands for the voltage that is being changed with potentiometer, while the R_{cs} is a constant resistance in the driver which is 50mΩ. If equation 4.5 is rearranged to get V_{ref} on the left side:

$$V_{ref} = 8 \cdot R_{cs} \cdot I_{max} \tag{4.6}$$ [3]

By adding the values that are known from the motor and chip:

$$V_{ref} = 8 \cdot 0.05 \cdot 0.4 \tag{4.7}$$

$$V_{ref} = 0.16V \tag{4.8}$$

According to the equation 4.6, one must have 0.16V displayed on the multimeter. Which would mean that the current is limited to 0.4A according to equation 4.5 The figure 4.12 display a driver zoomed in and a potentiometer circled red.

4.2.2 Wiring motor and driver

After making sure that motor wires are known and limiting current on the driver, one can move forward with connecting these into a circuit. Figure 4.13 illustrates a scheme of how to connect everything correctly. 1A, 1B is one pair of the wires that make up a coil, while 2A and 2B is the other pair that have been previously discovered. VDD is connected to the 5V output from Arduino, while GND is connected to any GND on the Arduino. STEP and DIR pins are connected to D10 and D9 pins accordingly. RST and SLP pins are interconnected. Since it's being set up for a full-step mode, MS1,MS2 and MS3 pins aren't used. When these pins aren't used, full step mode is enabled by a default settings. VMOT is a voltage which will supply a motor, in this case it should be 12V, because the motor has it as rated voltage. GND,which is next to the VMOT is connected to the GND of the external 12V supply. There is a bypass capacitor between VMOT and GND to lower the spikes in the power supply, which could cause inaccuracies.

[2] Source :https://www.pololu.com/product/1182
[3] Source :https://www.pololu.com/product/1182
[4] Source :https://www.pololu.com/product/1182

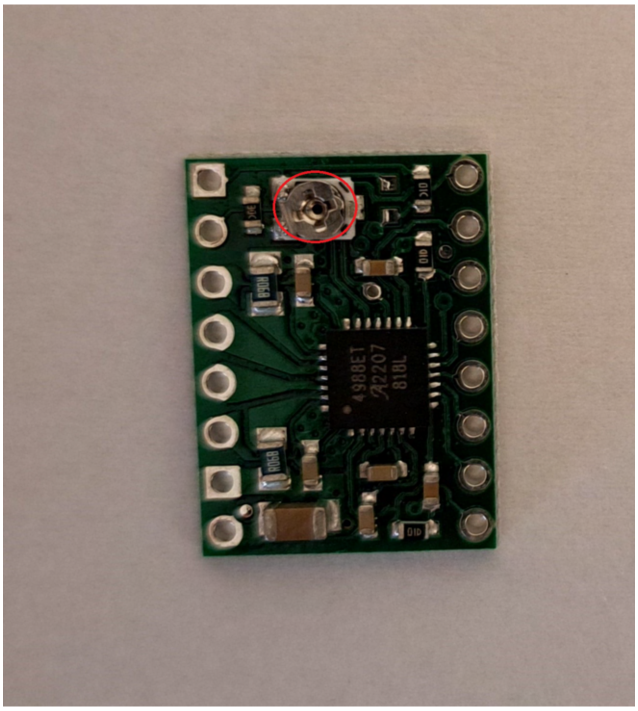

Figure 4.12: Potentiometer on the driver

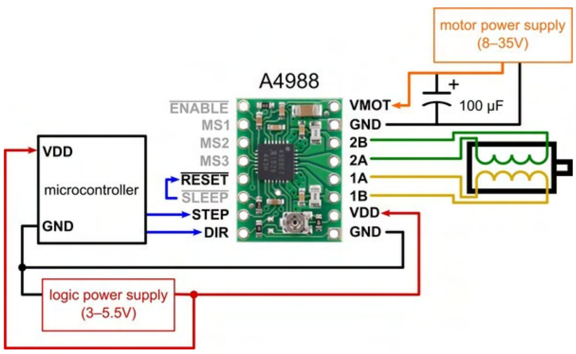

Figure 4.13: Illustrated Scheme of the wiring[4]

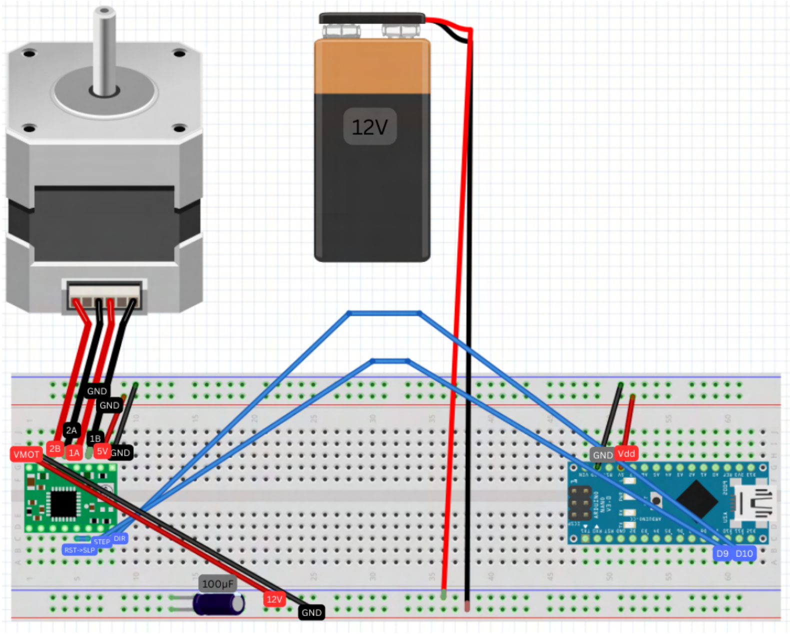

Figure 4.14: Diagram of the wiring

Figure 4.14 shows how all the things showing in 4.13 actually look like in practical example. There is an external 12 V DC power supply for the stepper motor. The power supply can be a battery or, as it is used in the project, an old adapter for phones. The capacitor must be an electrolytic one, it cannot be a ceramic, they're already embedded in the driver.

4.2.3 Software

Now that all of the hardware stuff is done, it's time to use software. In this case it is Arduino IDE. Since STEP and DIR pins were connected to the Arduino on pins D10 and D9, it also must be coded in that way in Arduino software. It is done by declaring a variable with integer number equal to what the pin was connected to on Arduino, which in this case is 10 and 9. In the function `setup`, those pins should be declared as an OUTPUT, since Arduino is sending signals to the driver. In the function `loop` one must input which direction is wanted. By choosing HIGH the motor will spin clockwise, while with LOW it will spin counter-clockwise. After the direction is chosen, a digital HIGH and LOW should be sent to the pin STEP with a delay in between those 2 signals. This makes a single step. The motor that is used in the project has 200 steps. If delay in between those HIGH and LOW signals is decreased so is the speed of the motor and vice versa. An important thing to note is that with the lower speed the vibrations will increase and in this project it is an

Figure 4.15: Damper[5]

important thing that could jeopardize the accuracy of the sensors used. The code reported in section A.1 shows the I/O setup for arduino.

Due to construction being made of plastic, which is quite lightweight, the vibrations that the stepper motor produces were quite loud and made the whole system unstable. With low speed, which causes more vibrations, the whole system was moving on the table by a bit. However, after increase the speed to lower the vibrations and installing a damper the vibrations were significantly lower. The stepper motor with a damper can be seen on the figure 4.15. As seen on the figure 4.15, it should be placed on top and screwed onto the motor. In order for damper to work, a stepper motor should not touch the ground, so it should be hanging, otherwise the damper is of no use.

4.3 IR Sensor setup

There are multiple IR SHARP sensor that can be used in the project. IR SHARP GP2Y0A41 and GP2Y0A51 are both a viable option. It was ,however,decided that GP2Y0A41 is a better option. The optimal way to see is to look at the figure 3.7 and figure 4.16. Considering the distance that is needed to measure in the project is from 6 to 12cm. Taking a look at the both graphs, one can see that figure 3.7

[5]Source:https://botland.store/other-elements/17909-vibration-damper-for-nema17-stepper-moto html

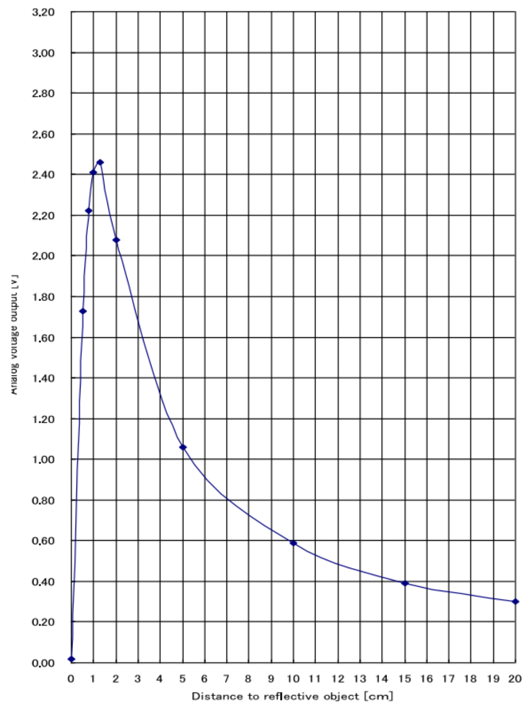

Figure 4.16: GP2Y0A51 graph [14]

which represents GP2Y0A41s graph, for the needed distance, the voltage varies from 2 to 1V. Taking a look at the figure 4.16, which represents GP2Y0A51s graph for the needed distance, the voltage varies from 0.9 to 0.5V. Which means that there is only 0.4V range in GP2Y0A51, compared to 1V of GP2Y0A41. Peak Voltage of both being similar. All this means that GP2Y0A41 has more sensitivity and is more accurate for distance that is used in the project, it has also been tested and better result have been achieved with that sensor.

An IR SHARP sensor have 3 wires: Vcc,GND and a wire for communication with Arduino. The scheme of how it should be connected is shown in figure 4.17. The connection is quite simple and if necessary a minimum 10μF by-pass capacitor should be added between mains, in case there a problems with stabilizing the voltage. Other then that, a GND of the sensor is connected to the GND of Arduino, while the Vcc is connected to Arduinos 5V. The third wire is connected to the analog pin 0 called A0 and is used to send data from sensor to Arduino.

4.3.1 Software

This section describes the software implementation that is in charge of controlling the hardware.

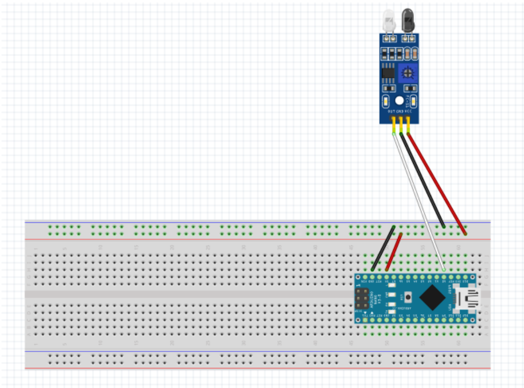

Figure 4.17: IR sensor scheme

IR sensors sends integer data to the Arduino. That data needs to be transformed into the distance, which is needed. The Arduino that is used works on 3.3V operating logic and it has 10 bit resolution. To transfer the raw data into voltage, one must use a formula 4.9.

$$V = data \cdot \frac{3.3}{1023} \tag{4.9}$$

The data that is received needs to be multiplied by $\frac{3.3}{1023}$, where 3.3 stands for operating voltage, while 1023 stands for $2^{10} - 1$ where the power 10 is the resolution of the Analog to Digital Converter (ADC) of the Arduino. The peak voltage would be a 1023,while the minimum voltage would be 0. Every value in between is divided by equal range. It's a typical ADC principal. Now, that voltage needs to be interpreted as distance and it should be done via the graphs that were already talked about in figure 3.7. The datasheet suggest using a equation 3.2, which was talked about in section 3.2 [8]. The equation 3.2 has a good approximation of the whole range of the sensor (4 - 30cm),however, since for this project the range that is needed is 6 -12cm, a equation 4.10 ,which better suits the data can be calculated. The equation 4.10 was done with the help of excel and its features, where one can feed it with values and then get a different version of graph and decide on one which suits it the best. The IR sensor was put on a desk and made unable to move. The ruler was put on the desk and a box which was moved from 6cm all the way to 12cm by an increasing 2mm after each measurement was taken multiple times and and average value was noted down. The table 4.1 shows the results of the measurements. `Voltage` column represents a voltage which was measured by a multimeter. The `Actual Distance` is a real distance that it should ideally measure. `Datasheet Distance` is a distance using a equation 3.2, while the `New Distance` is a distance

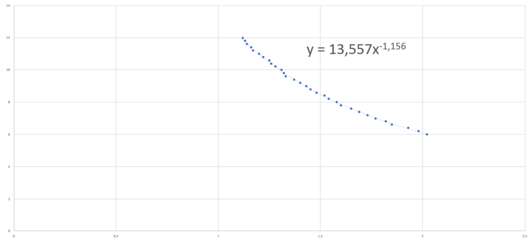

Figure 4.18: Voltage to distance graph new version

obtained with a new formula 4.10. The column `Difference`, shows the difference between ideal measurement and the measurement achieved by improving the graph with a new formula 4.10.

Looking at the data and choosing the best option of the graph, one achieves a new formula 4.10 which greatly increases the accuracy for the range that is going to be used in the project.

$$D = 13.557 \cdot V^{-1.156} \tag{4.10}$$

In the figure 4.18 one can see the voltage to distance graph, which was talked about and made with formula 4.10. Voltage is on X-axis, while distance is on Y-axis. The graph was made out of data in the table 4.1 and it's basically those data summarized. The dots represent a point where voltage is converted into distance, while the dashed line is a graph that tries to go about the dots in the most optimal way. After trying numerous graphs, it was settled that exponential graph gives the least amount of error. It is not perfect, some measurements are exact but most of them are missing a few decimals, it is simply the reality of using a low cost IR sensor.

Finally, the code was used to do the test and is generally used to run the IR sensor (see section A.2). First 2 lines enable so that the code can work with Arduino and a sensor, which is connected to the analog pin 0. A variable n is defined, which is used to tell how many samples, should IR sensor take as a group to calculate the average of it all. In the `loop`, the raw sensor data is multiplied by 0.00322580645 , which transform the raw data into voltage. That voltage is then multiplied by a formula 4.10 to get the distance. It is looped 50 times in this case, and an average value is calculated.

Voltage	Actual Distance	Datasheet Distance	New Distance	Difference (Actual - New)
2.02	6	6.43	6.019	-0.019
1.98	6.2	6.55	6.159	0.040
1.93	6.4	6.75	6.344	0.055
1.85	6.6	7.01	6.661	-0.061
1.82	6.8	7.15	6.788	0.011
1.77	7	7.37	7.009	-0.009
1.73	7.2	7.53	7.197	0.003
1.69	7.4	7.7	7.393	0.006
1.65	7.6	7.88	7.600	0
1.6	7.8	8.16	7.875	-0.075
1.58	8	8.24	7.990	-0.010
1.54	8.2	8.45	8.230	-0.030
1.52	8.4	8.55	8.355	0.045
1.48	8.6	8.77	8.616	-0.016
1.45	8.8	8.98	8.821	-0.022
1.43	9	9.09	8.964	0.036
1.4	9.2	9.29	9.186	0.014
1.37	9.4	9.45	9.418	-0.185
1.33	9.6	9.75	9.745	-0.146
1.32	9.8	9.89	9.830	-0.031
1.31	10	9.9	9.917	0.082
1.28	10.2	10.16	10.168	0.014
1.26	10.4	10.31	10.372	0.027
1.25	10.6	10.45	10.468	0.131
1.22	10.8	10.72	10.765	0.034
1.2	11	10.88	10.972	0.027
1.17	11.2	11.07	11.297	-0.098
1.16	11.4	11.24	11.410	-0.010
1.14	11.6	11.14	11.641	-0.041
1.13	11.8	11.63	11.760	0.040
1.12	12	11.62	11.881	0.119

Table 4.1: Measured distance data

Figure 4.19: XY motor fully assembled

4.4 Assembling 3D scanner

After making sure that every component is set up and ready to be worked with, its time to connected all the components and make a 3D scanner. One must place the stepper motors in the structure that was made for it that can be seen in figure 4.1. Since both motors are identical, both can be place at the either place. The motor that is a put on higher ground is called X and Y-axis motor, while the other is called Z axis motor. On X and Y-axis motor, one must place a damper, screw it down and place the front cover for the motor, seen in figure 4.2. On the shaft of the motor, one must place a double sided tape. The tape is there to reduce the vibrations. A coupler from figure 4.8 is placed on that shaft. A coupler is made in that way, that screw and nut are used to tighten the coupler around the shaft as needed. It's need to be tight enough so that shaft turns identically with the coupler. A coupler cannot slide on the shaft, while the shaft is not in a motion. However, it cannot be too tight, because it can break, since it's made from plastic. In the figure 4.19 one can see how it should look like once it's fully assembled.

Similar procedure is done on the Z-axis motor, where the only difference is that instead of turntable with coupler from figure 4.5, only a coupler is placed onto it. A 8mm lead screw is to be placed in the middle of a coupler and then it needs to be tightened. The lead screw should be about 15 to 20cm or as high as one wants to scan object. The height limit of the 3D scanner depends only on the height of the lead screw. In case one wants to scan an objects of height of 25cm one should have a screw of about 30cm, because the Z-axis is lower for 5 compared to the X and Y-axis. Looking at the figure 4.1 of the base of the structure, one can see the 2 slots for the rods right next to the Z-axis motor. These rods should be 8mm width and height 5mm higher than the lead screw. It needs to be 5cm higher, because the smooth rods are placed on to the ground, while the lead screw is placed on the coupler, which is connected to the motor which is again placed on the structure that

Figure 4.20: Bearing

is touching the ground, which results in 5cm height difference. Linear ball bearing from the figure 4.20 is to be placed on the smooth rods. 2 bearings are needed on each smooth rod, so 4 in total. Those 2 bearings at each rod are placed so close together so that they could act as one bigger bearing.

IR sensor holder from the figure 4.2 has a hole in the middle, which is meant for the lead screw. The lead screw is passed through the holder and the sides (as seen in figure 4.6) hold the linear ball bearings. The bearings need to be tightened so that they don't move. It is done with zip ties and also also premade holes for zip ties can be seen on figure 4.6. 2 zip ties per bearing are used, so 8 in total. IR sensor is to be placed on the front end of the IR sensor holder. It needs to tightened and made sure that it is not angled in anyway, because that will lead to problems with final product. A nut of 8mm needs to be placed in holder together with the lead screw A very small but extremely important piece is now used, which can be seen in figure 4.4. With screw and nut it is tightened on IR sensor holder. There is now a nut inside the holder and is pressed between a holder and a cover from the other side. This plays a crucial role, because now when the Z-axis motor is doing steps clockwise the IR sensor is going upwards(because it is placed on the front side of the holder) and similarly going downwards when the stepper motor is doing steps anti-clock wise. Bearings are there to make sure that the whole structure is standing still, unless there is a force being used in either direction. Regarding the X and Y-axis motor it can also spin in both directions clockwise and anti-clockwise, which result in the same movement transferred to the turntable, unlike in Z-axis motor where it results in going upwards or downwards. In the figure 4.21 one can see a fully assembled Z - axis motor with IR sensor. Looking from the backside in figure 4.22 one can see all

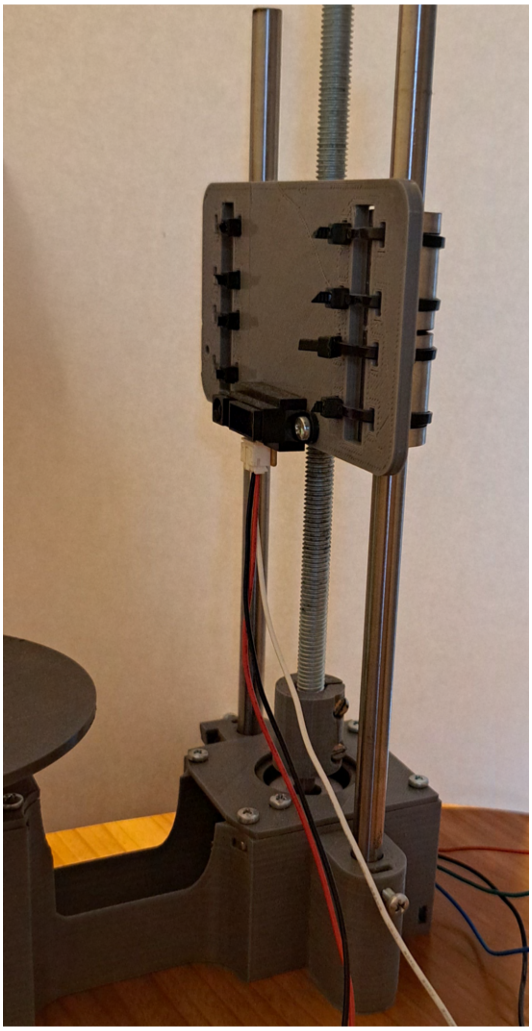

Figure 4.21: Z motor fully assembled front

the things explained in this section. A yellow ellipse represents a nut, while the red one shows bearings.

4.4.1 Software

After everything is assembled its time to explain the code that makes it all run. The code, reportedin in section A.3, is all of the code above the setup function in Arduino. It consists of many global variables:

- `steps_per_rotation_for_motor` is 200 because motor that is used does 200 steps per revolution or full circle.
- `Distance_from_sensor_to_the_center` is variable which represents a distance from the IR sensor to the center of the turning table.

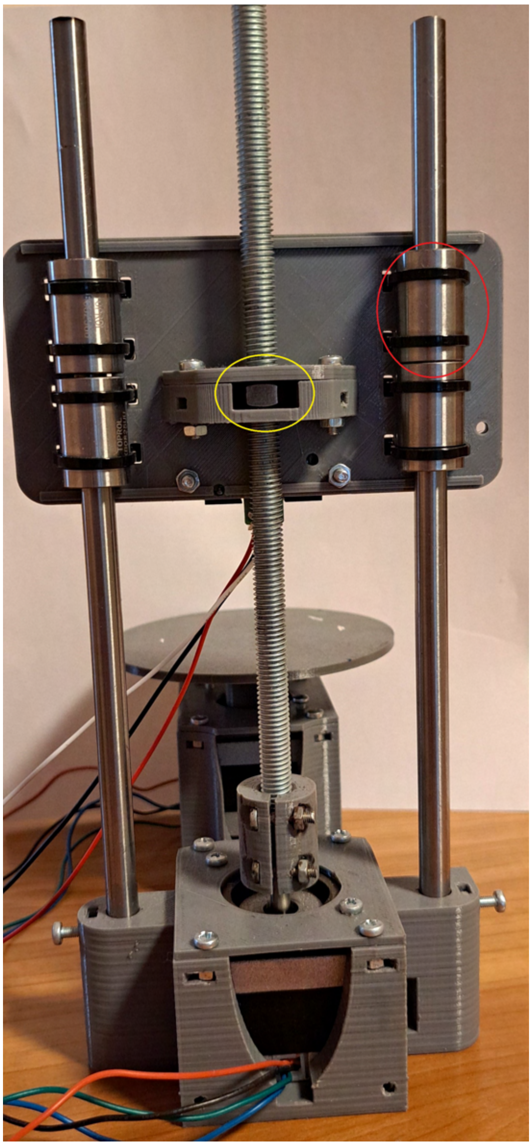

Figure 4.22: Z motor fully assembled back

- `lead_screw_rotations_per_cm` says how many revolutions or full circles are needed on the Z-axis motor for it to go upwards by a cm.
- `current_angle` keeps track of the angles of the item scanned in relation to IR sensor, while the `z_layer_height` is a variable that can be adjusted as wanted and it represents how often should one scan in height.

For example, in this case the variable is set to 0.05cm or 0.5mm. This means that the item is scanned by 360° and then its moved up by 0.5mm in this case. Digital pins 7,8,9,10 are used for motor drivers and STEP and DIR pins, while analog pin 0 is used for IR sensor communication. The `n` will be used to determine how many scans for a single point is going to be taken. In this case that number is 50, which means that at a single point, 50 scans are taken and then an average of it is later calculated.

The setup code reported in section A.3 comes afterwards. All the digital pins need to be enabled and a serial connection needs to be established with a small delay to wait for the python script to start. The python script will be writting down all the serial output of the program into a file. `steps_z_height` makes a conversion of the height in cm to height in steps. The height in cm was written in `z_layer_height`.

$$steps = height \cdot motor\_steps \cdot rotations\_per\_cm \tag{4.11}$$

Looking at the equation 4.11 where steps stands for amount of steps needed to achieve the height wanted, while motor steps is the amount of steps needed to make a revolution, in this case it is 200. Rotations_per_cm depends on the lead screw used and is measured on the screw. In this case it takes 8 revolutions on the screw for it to make a single cm upwards. Adding in all the numbers in equation **??** one gets number 80. That number means that 80 steps are needed on the Z - axis motor for it to go 0.05cm in one of the directions. 80 steps equals to 144° or 40% of a revolution.

$$steps = 0.05 \cdot 200 \cdot 8 \tag{4.12}$$

The loop code, reported in section A.5, shows the core of the program. First for loop does a revolution by performing 200 steps, but a step by step. A step is made by calling a `Make_a_step_xy_motor` which performs a single step. After that a `measure_distance` is called. In the function a sensor data is converted from bits to voltage and then from voltage to the distance. The average of n, or in this case 50 measurement is calculated. Another average called `post_filtering_average` is calculated. This average excludes any measurement which reads above 12cm, because this is a cheap version of IR sensor and there are sometimes incorrect readings which oscillate. These 2 averages are compared and if the difference is greater then 5% the measurement is considered invalid. This is all done for every step made for all of the 200 steps. After all 200 steps are done, it means that the object that was scanned made a 360° while being scanned every 1.8°. A `calculate_xy_and_print` is called, which calculates and prints to the serial monitor X and Y coordinates from the distance and angles, while Z coordinate is already known. Figure 4.23 will be used to explain how are the coordinates calculated. In the figure 4.23, black circle represents a turntable with a green objects on it, which is being scanned. Red ellipse is a scanner placed on gray holder. A distance d is the preset distance in program, which has been measured. It represents a distance from the sensor to the center of turning table. The distance d_m is a measured, so it represents the distance from the sensor to the object. These 2 distances are deducted. The angle is known, because stepper motors are used, which means that 1.8° per step. X calculation is shown in equation 4.13, while the Y's is shown in equation 4.14.

$$x = sin(\alpha) \cdot (d - d_m) \tag{4.13}$$

$$y = cos(\alpha) \cdot (d - d_m) \tag{4.14}$$

In case all 200 scans are invalid, it means that scanning procedure is done. Invalid is anything that is above 12cm or if the difference between 2 averages is greater than 5%. Since the scanner is made in a way that it cannot scan anything greater than 12cm, it means that if there is a measurement of 12cm or greater that it is simply wrong and therefore, if all 200 measurements are invalid in this sense it means that the scanner has gone above the object and there is no more need to scan and the procedure is stopped. The IR sensor goes back to it's initial position. However, if the scanning is not done, the IR sensor goes upwards by 0.05 cm by doing 80 clockwise steps on Z - axis motor. In order for the scanned object to look nicer when presented, a surface of the object is made from the data of the first revolution. Figure 4.24 depicts the process of how the surface is made. The most outer black circle has a radius of 10cm. The data that make up the circle are used and multiplied by 0.1, which makes the innermost gray circle. For the next step the brown circle is made of 2cm and it goes all the way until the original values.

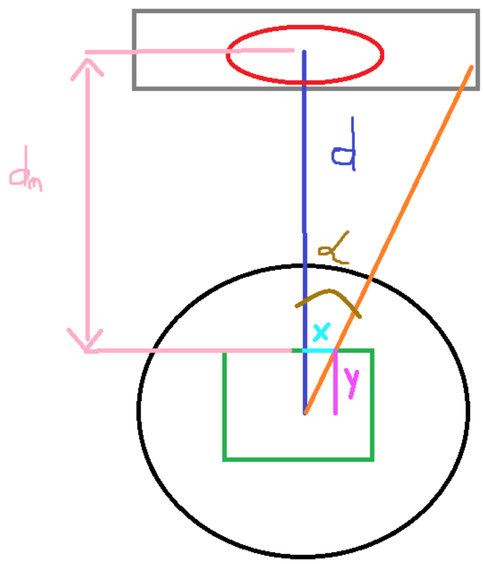

Figure 4.23: Calculation of coordinates

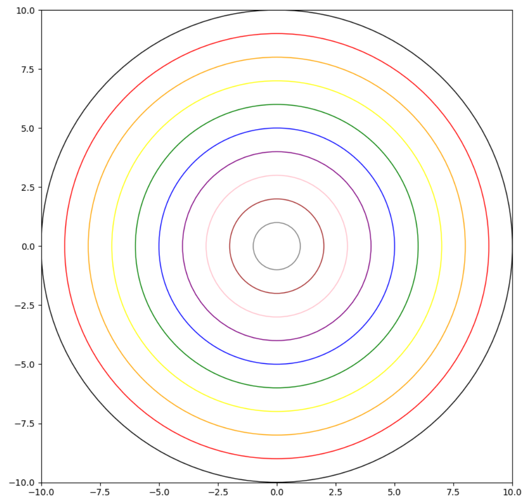

Figure 4.24: Process of making the surface

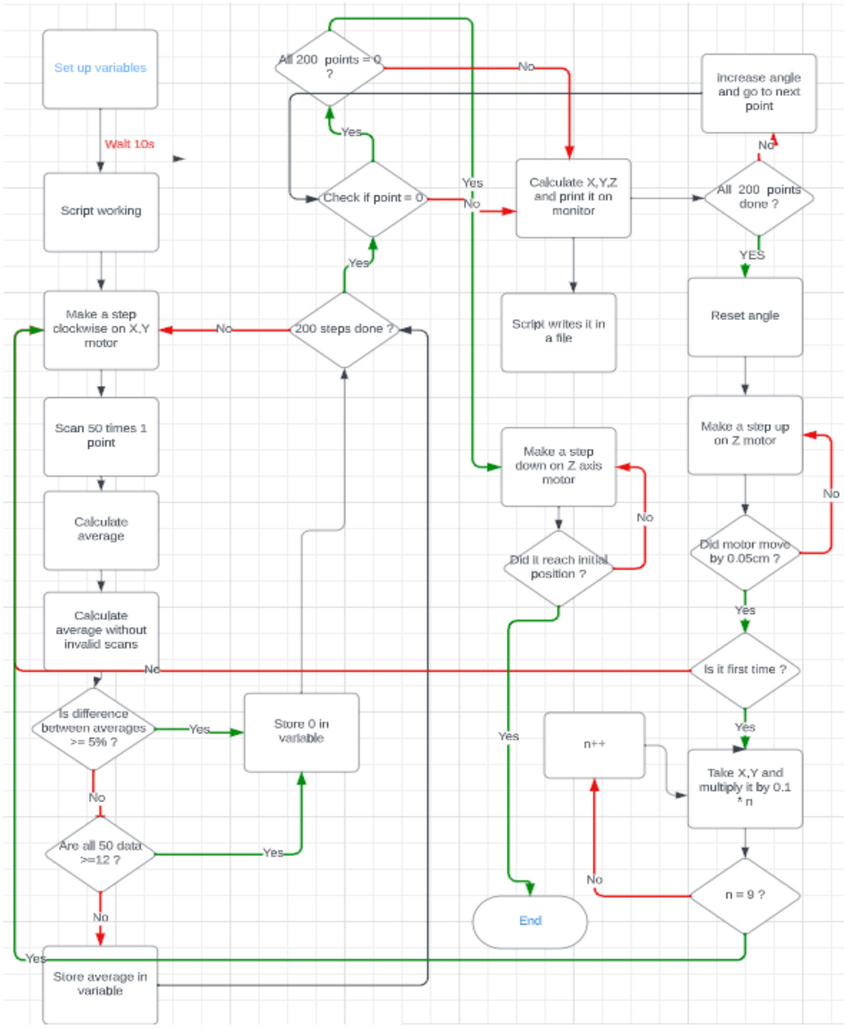

Figure 4.25: Flowchart representing the logic of the software

A python script (see section B.1) is simultaneously running and fetching all the data that is being printed on serial and writing it in a file. It knows which ones are X,Y and Z coordinates from the serial monitor and writes each trio in a row in a file.

Figure 4.25 depicts the flowchart of software. It describes everything that's been said in section 4.4.1 just made into a flowchart so that it is easier to understand what is happening at each point.

Chapter 5

Results

This chapter discusses the performance and the results achieved by the scanner.

5.1 Example of scannings

Multiple items will be scanned to show what it looks like scanned and the improved versions of scanned item too. Every individual object is deliberately selected in order to evaluate functions and capabilities of a 3D scanner. Items are made out of different material including : rubber, sponge and a ceramic. Some of them also including shinny surface, while others don't. The size also of a big importance to test out, how the 3D scanner behaves with different dimensions. Last thing taken into consideration when choosing the items is the complexity of an item and to see the exploits of a scanner. Additionally, regardless of items scanned, some disadvantages of a scanner of this type will be seen in the 3D models.

5.1.1 The mug

The first scanned item is a mug. The scan can be seen in figure 5.1, while the item that was scanned can be seen in figure 5.2. Looking at the figure 5.1 one can notice that the top of the mug is closed, while it should not be that way. This is due to an error of the IR sensor. The problem with the said sensor is that its working angle is too wide. By having a too wide angle it makes it detecting elements that are not there. This is an error that always occurs at the end of the scanning process and it seems to be 3 to 4 mm on average, regardless of the item's size. If information caused by an IR sensors error is deleted, the scan loses the closed top and increasingly portrays the real object as seen in the figure 5.3

Each dot in the figure is an average value of 50 samples in all scans, not just this one. The mug is made out of ceramic and has a shinny surface with multiple colors: white, yellow, pink, red, gray, orange and silver and blue. Even though there are multiple colors, there are also different shades of the same color. The dimensions of the item are :80 mm height and 68 mm diameter. If one includes a handle, at the part where handle is, the width is 92.5 mm. Interestingly enough, a figure 5.3 depicts a diameter of 66 mm, which results in an error of a 2 mm. This kind of error can be due to the size of an object or the fact that it has a polished, hence it has a reflective surface. This phenomenon occurs due to the inherent property of shiny surfaces to reflect infrared radiation rather than absorbing it. When an IR sensor emits infrared light and it reflects off a highly reflective surface, the sensor may detect reflections originating from multiple angles and sources, hence posing challenges in accurately discerning the genuine signal from the reflected signals. This phenomenon may lead to inaccurate data readings or measurements. The most reflective color on the mug is golden. The golden color has a line in the most upper part of the mug, which can be see in figure 5.2 of a mug. If one takes a look at that exact spot in the scanned figure 5.3, one can see that the line can be visible by being slightly displaced compared to other measurements taken. The line is moved inwards, which would mean that the more reflective the surface the less accurate the readings will be, by making it shorter. This observation provides a plausible explanation for the 2 mm discrepancy observed across the entirety of the mug. Last reason why the item was chosen is to see one of the disadvantages of having not only this scanner, but generally scanners of this type. When mug is scanned,it is placed on turntable and it rotates, while the IR sensor moves across the height. Problem is that there is only one IR sensor so when the item rotates, there are possibilities of a blind spots, which simply can not be seen due to the structure of a scanner. A good example would be a handle on the mug. The handle is seen and visible in the figure 5.3, however if one takes a look that there is a lack of a defined shape of the handle from the sides. This is because sensor simply never gets to read the data in the side area of a handle, due to the placement of a sensor. This is the general problem of these types of scanners. Another problem with scanners, which have turntables is that one can not scan the bottom of the item. This problem was partially solved, by making an artificial bottom of the item using its measurements, but its flaw will still be seen in the next examples. The quality of the scan is noteworthy as it closely resembles the scanned item. However, there is a slight decrease in accuracy attributed to the reflective surface of the item.

The quality of the scan is noteworthy as it closely resembles the scanned item.

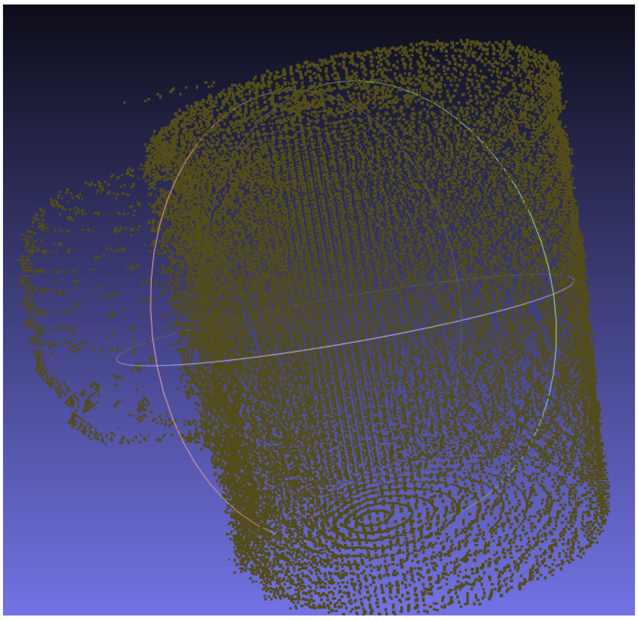

Figure 5.1: Scanned mug

Figure 5.2: Mug foto

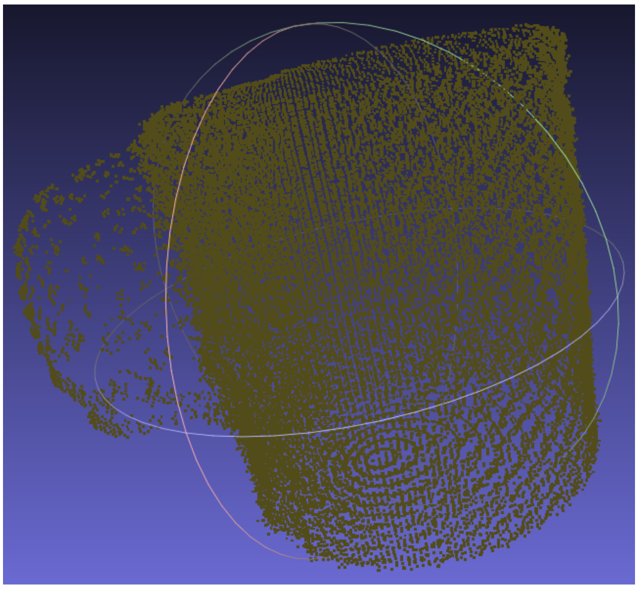

Figure 5.3: Scanned mug without error

However, there is a slight decrease in accuracy attributed to the reflective surface of the item.

5.1.2 The rubber ducks torso

Another scanned item is a rubber ducks torso. Scanned duck can be seen in figure 5.4, while the actual duck is seen in figure 5.5. In the event that data generated by an IR sensor error is removed, it is seen that the resulting models exhibit a closer resemblance to the real object compared in the figure As said before, even the duck is experiencing the same IR error where at the end of the scanned item it keeps getting some data for another 3 to 4 mm. The dimension of the duck are : 42 x 25 x 47.5 mm. Regarding size, it is much smaller than the mug and it does not have a shiny and reflective surface. The scanned object shows an error of 1 mm in width and height, which is acceptable. The duck is made out of rubber and is in only one color, pink. Purposely, it was taken only in a single color to see whether the color affects the accuracy of the scanner. The item was chosen to see if the sensor would be able to recognize a more complex structure. In the scanned figure 5.6 each shape of the object can be seen, a main body part, a tail and wings on both sides. From that standpoint it satisfies, however a bit of a definition of a wings is missing from behind, because there are blind spots where the scanner can hardly or not scan at all. The results of a scan are satisfactory with a decent accuracy.

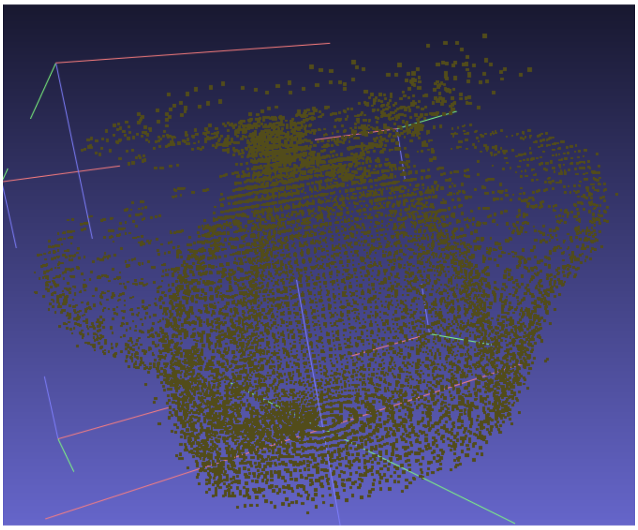

Figure 5.4: Ducks Torso scanned

Figure 5.5: Ducks Torso foto

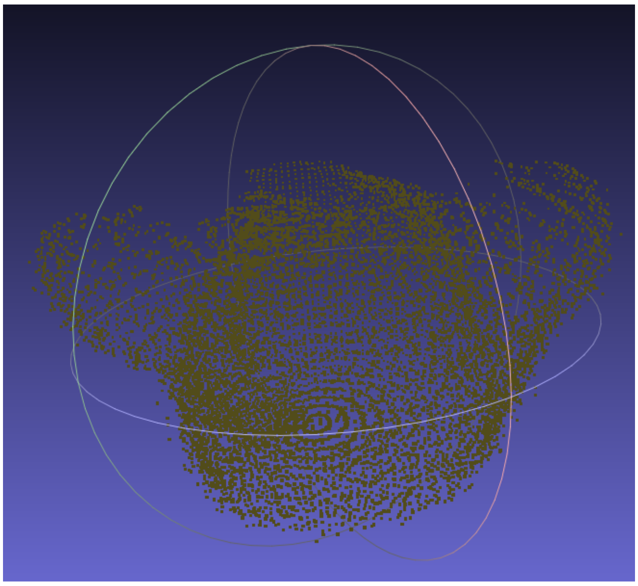

Figure 5.6: Scanned torso without error

5.1.3 The ball

Final showed scanned item is a ball. An item in real life can be see in figure 5.8, while the scanned is in the figure 5.7. When the data provided by an IR sensor error is eliminated, the resulting models demonstrate a higher degree of similarity to the actual object in comparison to the previously presented models in figure 5.9 A similar error can be seen, where towards the end of scanning a flat surface is formed. The ball is made out of sponge and it is smaller in dimension compared to the previous 2 items. Item was chosen, because it is has a diameter of 30 mm, has multiple colors and is generally considered to be a simple item to scan, however the sphere has evident wear and tear, such as multiple observable perforations and dents, which poses an interesting task for the scanner. Measuring at its widest point in the figure 5.7 one gets 29.6 mm, which would mean that there is an error of 0.4 mm. As seen in the figure 5.11 of model, one can deduct that these dents and perforations can be seen, but these also affected the accuracy of a model. Small protrusions can be seen in the model, which are the results of the accuracy of an IR sensor combined with defects of a sphere, which caused sudden changes in distance being mesaured by the sensor. As one can see in figure 5.9 the changes in the color of an objects does not affect it in any way. The ball is a great example of a disadvantage of scanners with turntables, which was talked about in Subsection 5.1.1. Instead of having a full sphere as one would expect, when looking at the bottom part of the sphere, one can see that at a certain point there is simply a horizontal cut. This is due to the fact that the scanned object is on the turntable an there is no possibilty to scan its bottom part. It's worth to notice, that the error is quite small and it is hard to see even on the model of 30 mm diameter when zoomed in on the scanned data in figure 5.9. The quality of the scan is commendable, while having an remarkable accuracy. The only drawback is the typical IR sensor error.

5.2 3D models

To make a 3D models of scanned items, scanned data with eliminated IR sensor error have to be used, which keeps scanning for additional 3 to 4 mm. For the purpose of seeing a potential if better sensor was used, the scanned data presented in section 5.1 are shown.

If one wants to use a simple filtering from meshlab to get an approximation of what a model would look like, it can be seen in figure 5.12 for a mug; figure 5.10 depicts a model for a duck, while the figure 5.11 illustrates the model of a ball. It

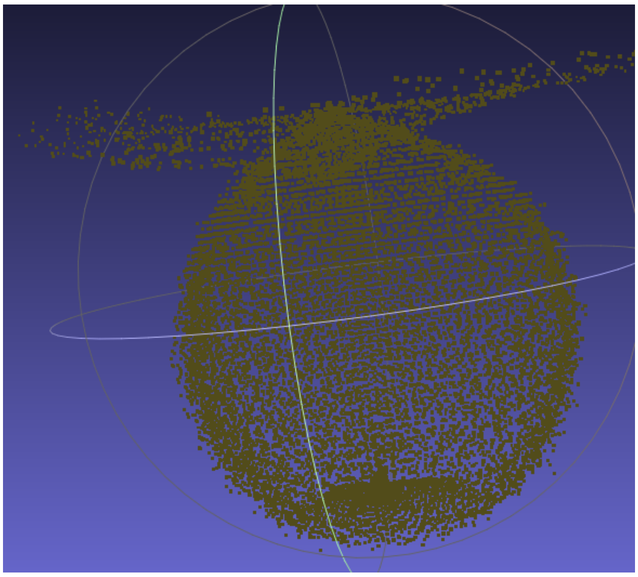

Figure 5.7: Scanned ball

Figure 5.8: Image of the scanned ball

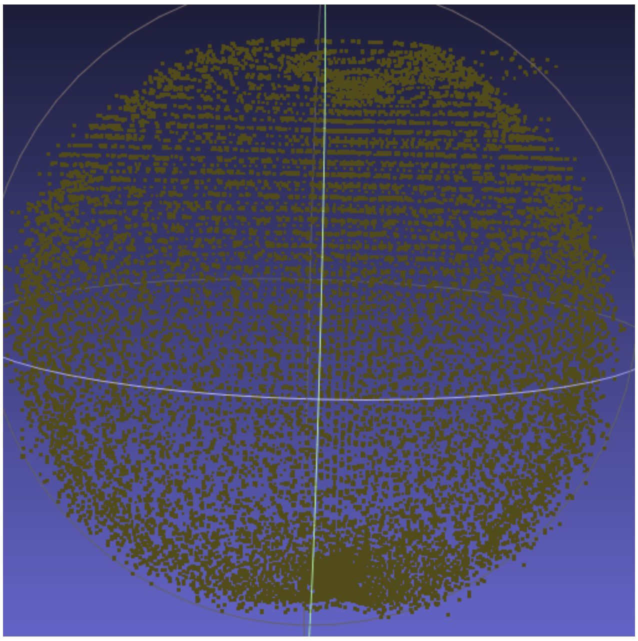

Figure 5.9: Scanned ball without error

is noteworthy to mention that the models are generated through the utilization of a basic model creation functionality within the software application meshlab. It is not perfect by any means, so some displacements seen are not necessarily from wrongful data, but from the software producing a model. Additionally, the models are significantly magnified, beyond the level of detail achievable in real-life objects. Consequently, even minor flaws may appear more significant than they actually are. The model of a mug 5.12 looks good, there are a few things that can seem weird, like the handle and the top part of the model. The software added a bit of the upper part, because of the algorithm that is used to create models from the data.

Where the algorithm performs poorly is in the handle. It was said that a shape of the handle from the sides is a bit lacking, and the software also performs poorly here and almost closing off the handle. It seems that it has a problem where the data is lacking or instantly ending, i.e., in the upper part. Overall the model is good and has an error of 2 mm.

The duck model in figure 5.10 has a better accuracy of a mug, having only 1 mm error. However, it is more complex shape than the mug and it does affect the final model of a duck. One can see that it has the rough shape of a duck; however, since it is zoomed in, it looks worse than it actually is. The actual dimension of a duck are only : $42 \times 25 \times 47.5$ mm.

The protrusions depicted in figure 5.10 exhibit a maximum deviation of 1 mm, a

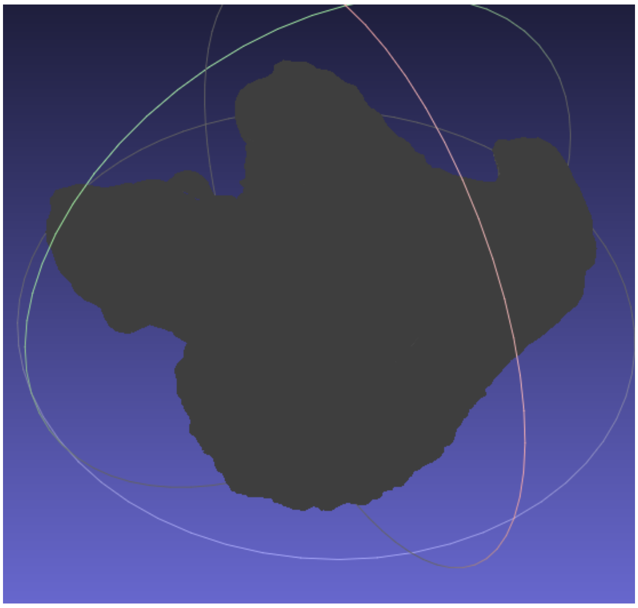

Figure 5.10: Model of a duck

rather little discrepancy. However, this slight mistake adversely affects the visual quality of the magnified model. The model in general is satisfactory. The best looking model of all of them is ball in the figure 5.11. The error is as small as 0.5 mm, while actually looking like a ball. Even though it is magnified model, only small protrusions can be seen, which is impressive. The ball is much smaller than the mug or the duck but seems to have the best model of them all.

Various insights can be derived from the aforementioned data relevant to the scanner. On of such insights is that the color does not affect the scanner in any way. It was tested on multiple colors ranging from dark to light colors with different shades. However, the surface and the material seem to affect the accuracy. Rubber and sponge had the best accuracy, while a higher deviations was seen in the mug, which was made out of ceramic and had a shinny and reflective surface. As one would expect, the simpler the item that was scanned, the better the final model is and vice versa.

The overall accuracy of the 3D scanner has surpassed the limits of the IR sensor, which was expected to have an accuracy of 1 cm. By adjusting the calibration of the sensor, and using the fact that a range of 6 to 12 cm is needed, a voltage to distance conversion was significantly improved. This resulted in achieving a 2 mm error in suboptimal conditions, such as reflective surface, which have been previously identified as having a negative effect on the IR sensors.

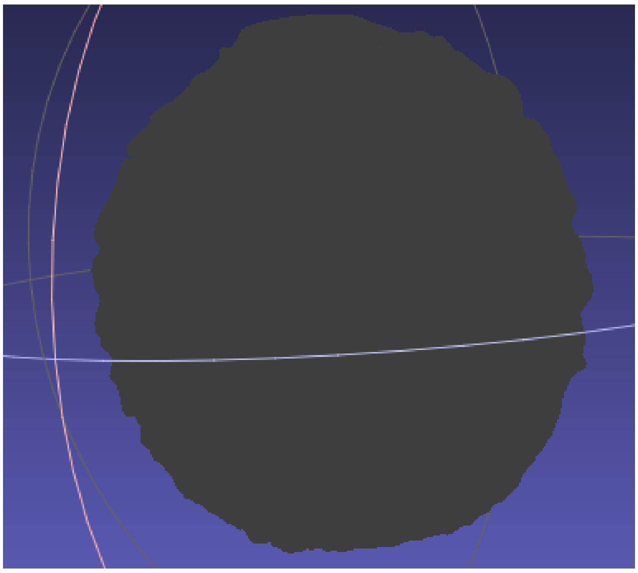

Figure 5.11: Model of a ball

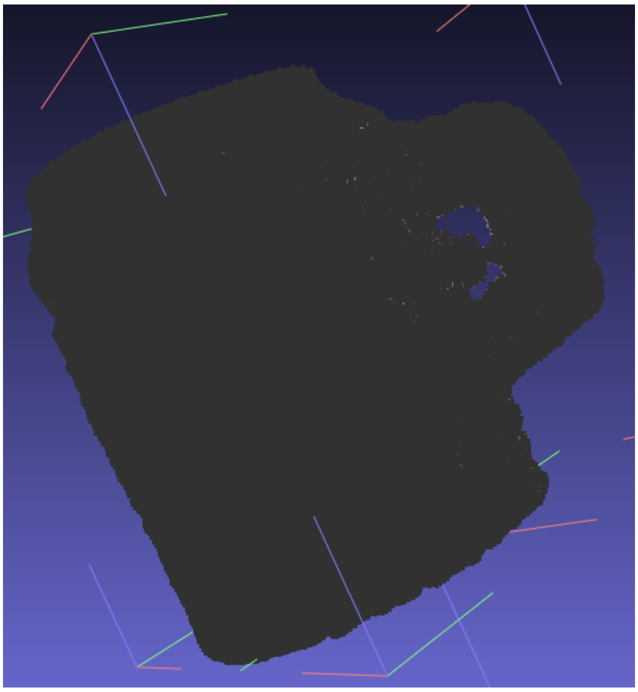

Figure 5.12: Model of a mug

Chapter 6

Conclusions

"Learn from yesterday, live for today, hope for tomorrow. The important thing is not to stop questioning."

Albert Einstein

The quote from well known physicist Albert Einstein fits perfectly in the conclusion of the thesis.

As it was initially planned to make a low cost 3D scanner, the goal was achieved and the price was lower than other scanners on the market. The whole idea behind it was well done and prepared, which included the whole perception on its working principle. The software and hardware were combined and made them properly working.

The only aspect that is lacking in the project is the quality of the IR sensor. Due to the fact that a low cost scanner was tried to be made and with limited budget, a SHARP IR GP2Y0A41 was used, which in the end seemed not the be fit enough to the purpose.

Looking at its datasheet, it seemed to fit all the criteria. However, since not everything is written in the datasheet, it wasn't accurate enough. If only a better version of an IR sensor was used the scanner shows a great potential.

This is the reason why a quote from the Einstein is a great fit here, many things were learned from the mistakes made, including not to use the certain IR sensor in this case. Comparing the prices between competitors and the function that they provide, the difference is not that significant as the price is. The scanner has a potential, because a simple change can results in the whole project to work in the similar rank a the competitors while being extremely affordable.

Subjectively it has even better structure, while having low requirements for it to be run.

Appendix A

Arduino code

A.1 Arduino Code for I/O Setup

A.2 IR Sensor Code

Listing A.2: IR Sensor Code

```
#include <SPI.h>
#define sensor A0
int n = 50;

void setup() {}

void loop() {
double data[n];
double average = 0;
int i = 0;

while (i < n) {
float volts = analogRead(sensor) * 0.00322580645;
data[i] = 13.557 * pow(volts, -1.156);
average += data[i];
++i;
}
average /= n;
}
```

A.3 Pre-setup code

Listing A.3: Pre-setup code

```
#include <SPI.h>
#define sensor A0
```

Listing A.1: Arduino Code for I/O Setup

```
#include <SPI.h>
int step_delay = 1000;

// I/O
int dir = 10;
int step = 9;

void setup() {
pinMode(dir, OUTPUT);
pinMode(step, OUTPUT);
}
void loop(){
digitalWrite(dir,HIGH);  // clockwise direction
digitalWrite(step,HIGH);    //make a step
delayMicroseconds(step_delay);
digitalWrite(step,LOW);
delayMicroseconds(step_delay);
}
```

```
//non adjustable variables
int steps_per_rotation_for_motor = 200; //Steps for a full circle
double distance_from_sensor_to_the_center = 9;
int lead_screw_rotations_per_cm = 8;
double degrees_per_step = 360 / steps_per_rotation_for_motor;

//will change values later on
double RADIANS_per_one_step = 0.0;
double current_angle = 0;
double z_layer_height = 0.05;

int steps_z_height = 0;
int z_loop = 0;
float z = 0;
double y = 0;
double x = 0;

int z_loops_done = 0;

int n = 50;
int step_delay = 1000; //in us

//I/O
int dir_z = 8;
int step_z = 7;
int dir_xy = 10;
int step_xy = 9;
```

A.4 setup code

Listing A.4: setup code

```
void setup() {
Serial.begin(115200);
delay(10000);  // Wait for the python script
pinMode(dir_xy, OUTPUT);
pinMode(step_xy, OUTPUT);
pinMode(dir_z, OUTPUT);
pinMode(step_z, OUTPUT);
RADIANS_per_one_step = (3.141592 / 180.0) * 1.8;
steps_z_height = z_layer_height *
steps_per_rotation_for_motor *
lead_screw_rotations_per_cm;
}
```

A.5 loop code

Listing A.5: loop code

```
void loop()
{
int is_scanning_done;
double data[200];
for(int i = 0 ; i < steps_per_rotation_for_motor; i++){
//make a full circle while scanning every step
Make_a_step_xy_motor();
data[i]=measure_distance();
}

if((is_scanning_done=calculate_xy_and_print(data))==1){
return_to_starting_position(z_loops_done);
// go back to the starting position( height)
delay(2000000);
}

current_angle=0;

//move sensor up by 0.05cm in z direction
while(z_loop < steps_z_height)
{
Make_a_step_z_motor_up();
z_loop = z_loop+1;
}

z = z + z_layer_height;
z_loop = 0;
z_loops_done++;
if(z_loops_done==1){
make_the_bottom_surface_flat(data);
}

}

double measure_distance(){
double data[n];
double average=0,post_filtering_avg=0;
```

```
int i=0,  number_of_overshoot=0;

while(i<n){
float volts = analogRead(sensor)*0.00322580645;
// value from sensor * (5/1024)   0.00322580645
// worked out from datasheet graph
data[i]= 13*pow(volts, -1);
average+=data[i];
++i;
}

average/=n;

for(i=0;i<n;i++){
if(data[i]>=12){
data[i]=0;
number_of_overshoot++;
}
post_filtering_avg+=data[i];
}

if(post_filtering_avg==0){

return 0;
}
else{
post_filtering_avg/=(n-number_of_overshoot);

if(post_filtering_avg >= average*1.05 ||
post_filtering_avg <= average*0.95){

return 0;
}
return post_filtering_avg;

}
}

int calculate_xy_and_print(double data[])
{
int j=0;

for(int i=0;i<steps_per_rotation_for_motor;i++){
if(data[i]==0){
j++;
}
else{

data[i]=distance_from_sensor_to_the_center-data[i];
y =  (cos(current_angle) * data[i]);
x =  (sin(current_angle) * data[i]);
Serial.println(data[i]);
Serial.println(String(x,3));
Serial.println(String(y,3));
Serial.println(String(z,2));

current_angle+=RADIANS_per_one_step;
}
```

```
}
if(j==steps_per_rotation_for_motor){ //check if whole
item has been scanned
return 1;
}
else{return 0;}
}

void Make_a_step_xy_motor(){

digitalWrite(dir_xy,HIGH);
digitalWrite(step_xy,HIGH);    //make a step
delayMicroseconds(step_delay);
digitalWrite(step_xy,LOW);
delayMicroseconds(step_delay);
}

void Make_a_step_z_motor_up(){
digitalWrite(dir_z,LOW);        //z_azis spin to right
digitalWrite(step_z,HIGH);      //z_azis make a step
delayMicroseconds(step_delay);
digitalWrite(step_z,LOW);
delayMicroseconds(step_delay);
}

void Make_a_step_z_motor_down(){
digitalWrite(dir_z,HIGH);        //z_azis spin to right
digitalWrite(step_z,HIGH);      //z_azis make a step
delayMicroseconds(step_delay);
digitalWrite(step_z,LOW);
delayMicroseconds(step_delay);
}
void return_to_starting_position(int z_loops_done){
int i,j;
z_loops_done/=2.5;
for(i=0;i<z_loops_done;i++){
for(j=0;j<steps_per_rotation_for_motor;j++){
Make_a_step_z_motor_down();

}
}
}
void make_the_bottom_surface_flat(double data[]){
int i,j,k;
double  distance[200],percentage2=0.1;

for(j=1;j<percentage2*90;j++){
for(k=0;k<steps_per_rotation_for_motor;k++){
distance[k]=data[k];
}
for(i=0;i<steps_per_rotation_for_motor;i++){
distance[i]*=(percentage2*j);
}
calculate_xyz_and_print_for_surface(distance);
}
}

void calculate_xyz_and_print_for_surface(double data[])
```

```
{

for(int i=0;i<steps_per_rotation_for_motor;i++){
y =  (cos(current_angle) * data[i]);
x =  (sin(current_angle) * data[i]);
Serial.println(String(x,3));
Serial.println(String(y,3));
Serial.println(String(z,2));
current_angle+=RADIANS_per_one_step;
}
}
```

Appendix B

Python script

Listing B.1: python

```
import serial
import time

# Configure the serial port
serial_port = "COM6"  # Update with the appropriate serial port
baud_rate = 115200  # Update with the baud rate matching the Arduino code

# Open the serial connection
ser = serial.Serial(serial_port, baud_rate)
print("Serial␣connection␣established.")

# Open the file for writing
file_path = "data.txt"  # Update with the desired file path
file = open(file_path, "w")
print("File␣opened␣for␣writing.")

last_receive_time = time.time()

try:
while True:
# Read three lines of data from the serial port
data = [ser.readline().decode().strip() for _ in range(3)]

# Combine the data into a single line with commas
line = ",".join(data)

if line:
# Write the line of data to the file
file.write(line + "\n")
file.flush()


# Update the last receive time
last_receive_time = time.time()

# Check if no data has been received for 10 seconds
if time.time() - last_receive_time > 10:
print("No␣data␣received␣for␣10␣seconds.␣Exiting.")
```

```
break

except KeyboardInterrupt:
# Close the file and serial connection on Ctrl+C
file.close()
ser.close()
print("File closed.")
print("Serial connection closed.")
```

Bibliography

[1] 3d scanner market - global forecast to 2028.

[2] Coordinate measuring machine (cmm) basics.

[3] British standard specification no. bs5000, part 60, 1982.

[4] Warsame Hassan Ali, Samir Ibrahim Abood, and Matthew N.O. Sadiku. *Fundamentals of Electric Machines: A Primer with MATLAB*. CRC Press, INC., 2019.

[5] Abraham Amole, M. O. Oyediran, O. O. Olusanya, W. A. Elegbede, A. T. Olusesi, and A. O. Adeleye. Design and implementation of a prototype active infrared sensor controlled automatic sliding door for mitigation of coronavirus disease 2019 (covid-19). 2:1–17, Oct. 2020.

[6] V.V. Athani. *Stepper motors*. New age international publishers, New Delhi, 1997.

[7] Mrinal Kanti Bhowmik, Kankan Saha, Sharmistha Majumder, Goutam Majumder, Ashim Saha, Aniruddha Nath Sarma, Debotosh Bhattacharjee, Dipak Kumar Basu, and Mita Nasipuri. Thermal infrared face recognition a biometric identification technique for robust security system. In Peter M. Corcoran, editor, *Reviews, Refinements and New Ideas in Face Recognition*, chapter 6. IntechOpen, Rijeka, 2011.

[8] Sharp Corporation. Gp2y0a41sk datasheet. Available online at https://global.sharp/products/device/lineup/data/pdf/datasheet/gp2y0a41sk_e.pdf.

[9] Seok-Hyun Ga, Hyun-Jung Cha, and Chan-Jong Kim. Adapting internet of things to arduino-based devices for low-cost remote sensing in school science learning environments. *International Journal of Online and Biomedical Engineering (iJOE)*, 17:4, 02 2021.

[10] Douglas W. Jones. Control of stepping motors, 1998.

[11] Boaz Lawnce, S Priyatharshini, Bharatha Sreeja G., and R Lavanya. Avr microcontroller based obstacle monitoring console for automobiles and industrial automation. 04 2007.

[12] Vaibhav Malav, Raushan Kumar Bhagat, Rahul Saini, and Udit Mamodiya. Bluetooth based home automation using arduino, 2019.

[13] Aamir Malik, T.-S Choi, and Humaira Nisar. *Depth Map and 3D Imaging Applications: Algorithms and Technologies*. 01 2011.

[14] Sharp Corporation. Gp2y0a51sk datasheet.

[15] Byunghun Song, Haksoo Choi, and Hyung Su Lee. Surveillance tracking system using passive infrared motion sensors in wireless sensor network. In *2008 International Conference on Information Networking*, pages 1–5, 2008.

[16] John-David Warren, Josh Adams, and Harald Molle. *Arduino for Robotics*, pages 51–82. 01 2011.

[17] Yi Wu, Saurabh Saxena, Yinjiao Xing, Youren Wang, Chuan Li, Winco Yung, and Michael Pecht. Analysis of manufacturing-induced defects and structural deformations in lithium-ion batteries using computed tomography. *Energies*, 11:925, 04 2018.