Group #9

Final Documentation

Short Range 3D Scanner

Project Members:

- Gabriel Akenten (EE)
- Franco Zavagnini (CpE)
- Naoufal Rihani (EE)
- Nirav Kotadia (CpE)

Table of Content

1.0 Executive Summary

Three-dimensional (3D) scanning is a method that obtains an objects 3D physical form and converts it into digital data to be displayed via an electronic data processor of some sort. 3D scanners are devices that have the ability to analyze the conditions of objects by collecting data points corresponding to the relative position and shape of the object of interest. The Collected 3D data, has many useful applications and can be outputted as many different formant that can be used by other existing program. Most 3D scanners already produced have the capability to scan industrially large objects or small objects, moving or stationary objects; but they are definitely not cheap. Our intended aim of this project is to be able to generate a design that incorporates a non-contact active scanner procedure which would use a line laser and optical sensors to be able to scan our object of interest.

Various technological techniques can be used to build a 3D scanning devices and each technology comes with its own limitations, advantages and costs. In particular, the over “3D laser Scanner” is designed to use a non-contact active scanner using an analog sensor, and a microcontroller. The scanner has an extending mechanism that moves round the object to be scanned in a circular format. Once the surface is completely scanned using the triangulation approach, a three dimensional mapping of the object will be extracted (point cloud) onto a software program that will make it possible to be able to make an alignment and mesh the point cloud to generate a polygon model (digital model). This will be displayed onto the monitor.

Although there are different techniques for the acquisition of 3d data, laser triangulation was selected because it can be implemented using “of the shelf” components; which will have a reflection on the total cost of this project. The general procedure in obtaining a generalized data; first of all, identifying the pixels of the image that are under the effect of the laser beam. Depending on how far the laser strikes a surface of the object, the laser line will appear at different places in the camera field of view. This procedure is achieved by subtracting two images, one taken with the laser turned on and other one without the laser being on. After the regulation of the laser being on and off, the resulting image created undergoes a threshold filtering in order to be used to identify the desired pixels. In the second step, the triangulation equations are solved and the virtual model is stored.

One of our many objectives will be effective interaction between the microcontroller, laser unit and the motors used. We consider using a servo for our rotational platform. Generally, servos offer better and adequate power and would be more compact than other available motors. A critical advantage of using servos comes with the ability to control or regulate to the desired speed.

Which make it ideal for open loop systems. Almost every electronic circuit has been designed to operate off some given supply voltage, which most of the time are assumed to be constant. A well designed voltage regulator would provide the required constant dc output voltage. The entire design would contain circuitry that would continuously hold the output voltage at the design value without worrying about any issues relating to changes in load current or input voltage , keeping in mind that the load current and input voltage are within the specified operating range.

1.1 Introduction

Nowadays, the use of three-dimensional (3D) scanners has become a widely used tool**.** These innovative devices are being introduced or made to overcome existed problems that seem either impossible or extremely harder to solve before. It only been possible in the recent years, because of the different types of sensors that had been developed to create a more commercial device which ended up playing a big role on designing 3D scanners. These devices gather information of an object or scene and represent these scenes graphically in a three dimensional unit using a different technique with extremely good accuracy. Three Dimensional scanners can be used in different disciplines, such as reverse engineering, they can also be used to create detailed automotive parts, virtual reality, visual effects, animation , sculpture, mold making, graphic designing, video games, 3D printing, robotic mechanics, medicine, forensics, dentistry, movies, museum, toy, architecture and manufacturing. Even though many commercial and sophisticated scanners already exist in the market such as Sense, Photosimile 5000, Photon , Dimbody, NeXTEngine and many others which are way too expensive, in the range of hundreds to thousand dollars, making them not affordable to hobbyist, general public and institutions with low economic resources.

The need to obtain a 3D model given a real object, a shape, or a surface is becoming more and more important in different fields and for different purposes. For this reason, 3D scanner builder is investing in new technologies able to see the entire shape of complex object without constraints in terms of object dimensions, locations, and materials. Because of that, many people have been working on projects to develop different goals in mind a low cost outcome, faster procedure, accurate, portable, etc. To give few examples manufacturing field need a relatively high accuracy scanner. There is like hundreds of 3D scanners aimed at the manufacturing market but here are some well-known, Hexagon and Konica-Minolta. For movie, Games, and entertainment accuracy is probably less important here; probably more interested in capturing the shapes and texture of objects. Some of well know scanner for this field are XYZRGB and Cyberware. LDI and Nikon Metrology used in quality inspection because high accuracy critical. Cadent iTero are used for medicine and dental practice. There are couples of ways in order to obtain a three- Dimensional map of an object or scenes. If there is a need of classification of the type of scanner being used in these scanning approaches, they can be divided into two categories: The contact and non-contact scanner.

Contact systems, which probe the object you scanning through physical contact which is mounted to floor or roof; tip of the arm to be able to slide along each of the dimension. It's equipped with high precision sensors, to pin down the position taken from the surface to join the scene. These type of scanners could create some physical constraint that could be dangerous and dames the object being scanned. To be able to avoid these kinds of scenarios from happening, some scanners are generally made with the idea touch free scanner. These are generally known as non-contact 3D scanning systems. They replace the physical probe with laser and can be used in many different ways. Laser scanners system uses sensor (cameras) held within the scanning head to capture the image of the object using triangulation techniques, resulting in a point cloud system. These types of technology turn out useful for small range and long range. Long range scanners system are known for being used for scanning large objects such as planes, buildings, rooms and bridges. This technique allows a user to capture data with very good accuracy**.**

There is one more method, called the photogrammetric, which are widely used in phone cameras to be able to create a three dimensional image. The photogrammetric system approach works by taking multiple images of the object of interest and using some reference points from each differing angle from which the images were taken producing scan data.(ex. Google Nexus camera app). There are couples of new trend approaches where the objective in mind is making a handy scanner (portable scanner) which uses laser trackers or with the use of markers instead. This type of new scanner approaches have the advantage to get the entire object shape without changing the object of interest's relative position at any given time, but moving the scanning device without the involvement of long software alignments wouldn't work . These new scanning techniques might also require the incorporation of an ideal external tracking sensor such as MARG (Magnetic Angular Rate Gravity) which would make the entire project less cost effective. So how do you make cost effective 3D scanners?

2.0 Project Description

2.1 Project Motivation and Goals

Our goal for this project is to be able to design a low-cost scanning system and implement a laser to generate a 3D digital model of an object .Our project will be presenting a hardware and software system for digitizing the shape and color of our chosen object. By the end of our design, we should be complete the scanning a 3D object in a given amount of time and accurately display the generated 3D model in MATLAB and final outcome should be comparatively cheaper than commercially available 3D scanners. Data acquisition from the scanning point will be a critical aspect of this design since this will enable the plotting needed to be developed. The scanner will consist of a motor driven turntable, laser, camera, and control logic. During operation, the turntable will provide a rotational movement, to allow the camera to capture and generate an image data of the object from every angle. The results from the system will be carefully monitored and displayed so that an end user could be able to tell if the device is efficiently performing to its desired specifications. These outcome images will be stored on our computer which then will process those images and finally create a 3D model in MATLAB. The group will gain knowledge and experience from this project which will be beneficial since 3D applications are very new on the market and future employers might take it into consideration. Other relevant aspects of the design will be broadening our scope of knowledge.

2.2 Objectives

We are dedicated to achieving an efficient successful project by applying all our academic standpoint of view and utilizing the skills acquired throughout the entire UCF Electrical and Computer Engineering curriculum and showcase them in such a way that, it reflects on the talent as a group. After careful analysis, some of our objectives will include the following.

- Accurate point cloud representation of modelled object.
- Limited coding for our computer vision approach.
- Creating room for re-experimentation and error margin from the field programmable gate array
- To be able to achieve the maximum overall clock rate from our microcontroller.
- To effectively power our stepper motor from our control signal.
- To adequately achieve raw data images from laser and camera and the final translation of files.
- Triangular geometric approach to measure dimension of object to be accurate.
- Successful laser transformation to stripe of flight to enable movability and adjustments.
- Structured light mechanism with minimal error.
- Correctly matching a stripe order of the reflected superposition patterns for calibration
- Adequately supplying all the required power ratings, currents and voltages to all components of the design.
- Efficiency should be at least 40%

2.3 Project Specifications and Requirements

Brief specifications, outlined below will be some of the ideal guidelines that this project will be taken into consideration to be able to generate our final data for adequate modelling and processing. Each component forming part of an overall successful design will be given a systematic and thorough approach before their selections are made. These requirements were made possible by each individual member of the group being dedicated and including researching to that particular section.

Microcontroller

- Range of 3v to 5v supply voltage
- Maximum overall clock rate not to exceed 20MHz
- Controlling information signals required including rotational information.
- Send data generated to computer for processing
- Enough pins for input
- The output of the motor control supplies the input to the motor drivers. This determines how the motor will step through each of the rotation

DC Power Supply

- Capable of supplying adequate power requirements for the entire setup.
- Regulation to our desired DC values

Motor Module

- Motor to provide enough power and rotation needed. Supply voltage at 12V and various angle rotational movements

Laser Module

- Be able to spread the laser beam into a vertical line with a measured distance away.
- Adequately collect laser beams which were reflected for data processing on our computer.

Processing Module

- Carry out calculations necessary for a scan to be completed including still output from the camera
- Point cloud verification with known input.
- Model synthesizing developmental analysis.

Software Module

- Obtain and process the scanned object from laser/camera.
- Render and model the scanned object in the data representation software.
- Pass the scanned object through all the algorithm to best represent the object mathematically as well as geometrically as real as possible.

3.0 Research

3.1 3D scanning approaches

Most of the scanners can be categorized by two main categories: contact based and non-contact based.

3.1.1 Contact Based

Contact based scanners are in direct contact with the surface of the object to be scanned. A probe is used to estimate the shape by recording the displacement of the probe as it slide across the solid surface. While effective, this type of contact based methods can harm fragile object, they require long periods of time to build an accurate 3D model. If they move past there is good chance that it will damage the object.

3.1.2 Non-Contact based

Non-contact based scanners can further be sub-divided into passive and active scanners. While both scanners relay on one or multiple light sources to reconstruct the shape of the object, passive scanner do not need to actively control the light source, instead they rely entirely on surrounding light. Scanning without contact allows high speed scanning while avoiding physically damaging the object, which is important for instance while scanning antiques stuff.

Passive scanning -Famous example of this type of scanning is how old day 3D movie work with blue and red light. It’s known as stereoscopic imaging. In this approach, two cameras are used to mimic the human visual eyes. Knowing how position of the cameras it is possible for each camera system to independently establish an equation of a light corresponding to point in space. By calculation the intersection of those resulting two camera rays, the position of the specified point can be extracted. This concept can even be extended by adding more cameras. This technique is quite challenging especially when scanning flat or periodic textures. This problem knows as correspondence problem but, now day some of the portable 3D scanner used this technician.

Active Scanning -Active optical scanners avoid the correspondence problem by replacing one of the two cameras from stereoscopic imaging with a laser or a projector, which we see how most of the scanner now days uses. In comparison to non-contact and passive methods active scanner is often sensitive to surface material properties like reflection and absorption. Using line laser scanners create a planer sheet of light (vertical or horizontal line) which defines a plane on to object. This line of light is then mechanically swept across the surface of the object to be scanned. Then depth is calculated how using basic of triangulation. Other way to do is to project structured light (multiple lines at same time) which will used to eliminate the mechanical motion to translate the line across the surface and it is faster to create the 3D model. Another approach is time-of- flight rangefinders. Which is used to create 3D model of bigger area like building and stuff. They estimate the distance to a surface from a single center of projection, so holes do not appears. Since the speed of light is known, it is possible to estimate the depth by measuring the elapsed time between emitting and receiving a pulse of light. However, the depth resolution and accuracy of system remain below that of laser ranger scanners and structured lighting.

3.2 Existing Similar Projects

Among all the present day scanning technologies, there is really expensive scanner that give the good model with good accuracy such as Artec 3D Scanners and Next Engine. And there is really affordable ways to make 3D scanner that provide really poor accuracy such as David Scanner. But Line-laser range scanning offers the best combination of accuracy, affordability, scanning speed, and simplicity. This technology is also used by professional scanners such as the MakerBot 3D scanners. In this section I present you with list of affordable 3D scanners and throw some light on a numbers on existing product.

3.2.1 NextEngine

According to *nextengine.com* their product delivers high reliability at low price. NextEngine 3D scanner is capable of 400DPI resolution, .005in dimensional accuracy, and a 22.5in x 166.75in field size. It required some experimentation to get optimal scan, and failure to learn can return spotty results. It also has short scan times (50,000 point/sec throughput), rapidly available scan results and scan controls that are easy to manipulate. Multiple views of the object also provide feedback on scan density and color. The scanned object can be seen with textured surface, mesh and point cloud. In the end typical small model has 250,000 points. And cost about $3000 and still be said to cost ten times less to other equally good 3D scanner. It contains two arrays of four solid state laser, two 3.0 megapixel cameras, and two lights for image capture. The arrayed lasers and sensors use in the scanner is MultiStrip Laser Technology. “RGB color texture capture” to produce texture on 3D model, and one rotary servo positioner is control by software. It required separate power for the scanner, which is connected to the computer with USB.[1]

NextEngine scanner has positioning plate which can handle objects weigh up to twenty pounds and measuring up to 8 inches in diameter and 11 inches in height. It also can scan bigger object but it has to do piece by piece which will be assembled into computer software. Software that comes with NextEngine (ScanStudio) is capable of doing three things: to focus and define the scan parameters, to align multiple scans, and to post-process align scanned data. Software environment is very simple: 3D geometry is shown in the main window; all the tools are accessed using the toolbar at top, and at bottom is a library of scans. It also shows live video image from the camera in the scanner, so during scan it's possible to adjust settings according to needs. It is also possible to measure distance of object in software. NextEngine can scan a limited size object only, so to make it easy they created markers which will helps stitching multiple scans together easily. Depending on the scope, resolution and number of divisions selected full scan may can take hour. ScanStudio outputs 3D geometry files which can be opened on CAD programs or can be use for 3D printing.

Fig 3.2.1(1) NextEngine 3D scanner
(Credit to NextEngine.com)

3.2.2 Artec Spider

Artect 3D scanners are being used by Hyundai motor Europe to visualize and modify automobile seats for new cars from Hyundai. Artec 3D scanner is one of the high-end scanner; far as you can go with today's technology to make one of the best 3D scanner in every imaginable way. Best for quick, textured and accurate scan. That doesn't require markers, manual alignment or calibration. It capture quickly in high resolution and color. It is similar to a video camera that can capture 3D instead of 2D. All the processing happens in real-time. Artec Spider is designed to scan big and small objects with complex geometry, sharp edges and thin line. Scan such objects like human ears, key or coins. Also good for crating highly – detail CG with resolution of 1.3 megapixels. But unlike NextEngine this is hand held device and does not need fix flat form, that mean no restriction on size of the object. Also this type of scanner are better for museum art work where its heavy to move are worry that you may damaging the work. Unlike most 3D scanner out there Spider has been developed for scanning objects with many small details, such printed circuit board (PCBs). It's mostly use by big company for medicine, automotive, and aerospace.

Artect Spider is capable of scanning a coin surface with fine accuracy. If we have to measure in number then here are few from *artec3D.com*: 3D resoling up 0.5 mm, 3D point accuracy up to 0.1 mm, 3D accuracy over distance up to 3%, texture resolution 1.3 MP, colors 24bpp, working distance 1 meter, video frame rate 16 fps, data acquisition speed up 2 million points/sec, power consumption 12V, and it uses a flash bulb (not laser) [2]. Biggest use is in medical and biomechanical research because of the high quality and color. But starting prize is $15,000 and up. Spider work on battery and send data to computer wirelessly so there is no problem with wire getting in way of scanning like Sense.

Figure 3.2.2(1) Artec Spider
(Permission from artect3d.com)

Software that comes with scanner is call "Artec Studio" capable of live scanning preview, point cloud filtering, merging, editing, sampling, automatic and manual image controls, continuous shooting modes, meshing, sharp edges refinement, auto save and crash restore etc. It also required high end processor and GPU in your computer to run properly. Software its self-cost $650. ArtecStudio also supports the other low cost 3D scanners like Sence, and Microsoft Kinect. Microsoft Kinect is not a 3D scanner but technology is the same. And can be used to create 3D model.

3.2.3 MakerBot

The MakerBot 3D scanner (Digitizer) is desktop 3D scanner. The company is already well known for 3D printers, so they wanted to build a 3D scanner to go along with its 3D printers. Commercially available 3D scanners are quite specialized. MakerBot CEO BrePettie said “We wanted to make something for people to scan things on the Replicator 2(3D printer)”. Company goal was to make 3D Xerox machine. The Digitizer outputs standard 3D file formats so that it’s easy can improve, shape, mold, twist, animate, and transform objects in third party 3D modeling program. According to *makerbot.com “*Desktop 3D scanner’s easy to use, yet sophisticated software creates clean, watertight 3D model that are ready to 3D print. You don’t need any design or 3D modeling skills to get started and it all happens in just minutes”. But acceding to customer review of product based on Amazon; I realize the digitizer does have difficulties scanning sharp rectangular shapes. And machine works best with work with light, matte objects such as ceramics, clays and non-glossy plastics. Also most of the scanned aren’t what they claim to have. And cost around a thousand dollars.

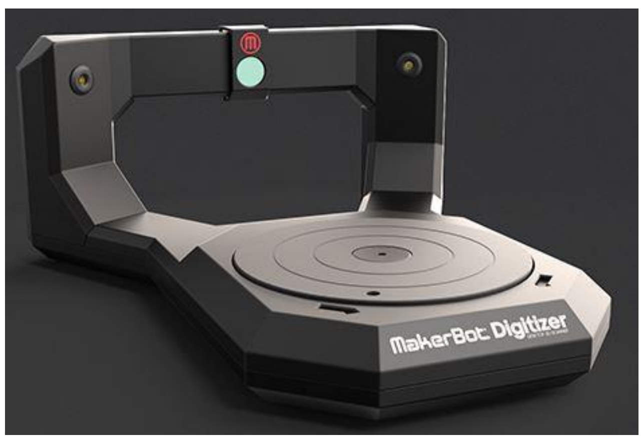


Figure 3.2.3(1) MakerBot Digitizer
(Credit to makerbot.com)

The digitizer can scan objects up to 8 inches across and 8 inches tall. Dimensional accuracy is 2mm. Detail resolution is 0.5mm. Triangles per 3d model roughly 200K.It can handle objects up to 6.6 pound, and take around 10 min to scan an object to a digital file [3]. Digitizer has similar design to what we try to build on our scanner. To put it in general it has a turntable connected with stepper motor, 1.3 megapixel CMOS image sensor and two line laser generators. The Object is placed in the center of a turntable, which slowly rotates as a laser is pointed right down the center of object. Reflected light, creates a point map of the object. It does two scans on second scans its scene from different angle and then both image are combined in software that comes with it (MakerWare) to create the 3D scene.

MakerWare give the life feed of scans well as mesh layout showing you how much of the project has been scanned. After finishing scene MakerWare will walk you through sliding away the camera filter so you take a photo (.jpg) of your scanned object; so it's easily share on social network. There is no patching or repairing of the initial scanned model required. MakerWare will output a regular 3D design file which can be upload to any third party software to edit.

Another option is to send the digital files to a 3D printing company like Shapeways or Sculpteco; companies are famous for making 3D copies in finer materials including ceramic and even silver. This opens new door for the artists who doesn't know how to use 3D design software but this can put more value in to their work. Such as jewelry, sculpture made of silvers. This process is not cheap but it be easy for people with unique talent show their work. Finally, this a robust and reliable 3D scanner available for an affordable price; however, our intent is to design and construct a cheaper 3D scanning device so that it can be used as the same as this one.

3.2.4 Sense

Not everyone afford Artect 3D Scanner, but to scan large objects you have to have portable scanner and it have to be high-quality 3D models with reasonable prize. That is where Sense 3D scanner comes in at $400. It is not 100% portable because while scanning you have to be connected to computer with long USB cable. It is kind of based on first generation Microsoft Kinect camera for the Xbox.

Sense can scan object from 0.35 m to 3 m, with accuracy of 0.9mm. The amount of time it takes to scan something varies depending on the size and detail of the object. Software can give live tracing of object during the scan. Also can solidify the object, cutting, smoothing the 3d model, cutting etc… Amazing thing is you can buy used Kinect for $50 and it will work with software like ReconstuctMe, to create same effects.

Figure 3.2.4(1) Sense 3D Scanner
(Credit to igo3d.com)

3.2.5 Microsoft Kinect

In 2011, Microsoft released the Kinect, a device for motion and body sensing. That supposes to replace the normal handheld game controller for Xbox 360. Also recognizing and tracking player identity. Besides a set of other sensors, the Kinect also includes projectors and cameras which are used for depth detection and color; however, this setting can also can be used to design 3D scanner. There are more accurate methods, of capturing data in 3D way; what makes Kinect attractive is its prize $100-$150. Since its Microsoft product with work any pc computer.

The Kinect works by projecting an array of infrared dots into the space immediately in front of it. The CMOS sensor tracks how people in its field of vision reflect the infrared beams back. A color camera and microphones give the feedback for quick response. Kinect works best at distance of 6 to 8 feet and it has accuracy of around 10 mm. Since Kinect work good with mid rang distance its use for scanning big thing like human body or sofa rather than use for precise short-range 3D scanning.

There are some Open Source software written for Kinect to use as 3D scanner like ReconstructMe; and other software that are not open source also supports like ArtecStudio and Skanect. ArtecSudio gives two weeks trials so people can try out their software without buying hardware before they spends hundreds of dollars on product. Artec Group also offers an online service call Shapify that lets users make accurate figurines of themselves and others. Shapify provide website that allows to submit one person scan. This can be created with MS Kinect; which is then 3D printed and home delivered by mail.

Open Source software ReconstructMeprovide basic functionality like start or pause scan process, turn the data into 3D polygon model also support few output format. You have to use addition software to edit the 3D models but this is low-cost 3D scanner for engineer and hobbyist. ReconstructMe also work with DAVID 3D scanner.

3.2.6 David Scanner

DAVID –laser scanner was designed to scan simply using a line laser, a webcam and a reference scanning corner. DAVID Scanner is based on depth-sensing technologies. Triangulating laser scanner project laser line on the object and record the distance. David scanner is not run by big company instead it run by group call David Vision System(DVS) they don’t sell pre built 3D scanner, instead they sell do it yourself kit to make the scanner. Which include just a webcam, mini tripod, handheld line laser, cardboard calibration backdrops, and scanning software provided on flash drive. To make the scanner you place an object in front of the calibration board and shine the laser on it from various angles. The webcam captures the reflected laser light, and the software assembles the 3D model. The whole idea behind the DVS was to make 3D scanner like hobbyist and affordable. And it also possible not to buy kit and make your own line laser and better camera to come up with better result. With kit equipment it cans scene size 10mm to 400mm in. but it's possible to scene big object by scaling the background marks. The kit yield fairly detailed scans with accuracy with 0.5% of the object size, but not good enough to capture surface texture. It roughly takes 40 sec. To make whole 360 model of the object user needs to turn the object manually between multiple scans then stilts together with software like 'Shape Fusion'. There is no guaranty how the scanner will turn out because individual have to hold line laser and has to point the right way. Still you can get scans with a surprising precision simply scanning by hand. David scanners sell their kit at $550 but it possible to create your own. You do not need additional hardware, but it might help to increase speed, accuracy and to complete jobs that you cannot do by freehand scanning. There are open hardware project base on David scanner to make it even cheaper. So everybody can build the system without any special skills, since its open hardware everybody have the choice to build it brick by brick. And one can freely use the technical designs, expends or modify them as one like. There is also lot of other hardware electronics that can use such as Arduino. There are lots of resources out there for this kind of hardware: how to use them, how to program them, it can create lot of possibilities.

Figure 3.2.6(1) DAVID 3D Scanner
(Permission from rob.cs.to-bs.de)

To make precise 3D model David Scanner sell second kit which is more expensive and use it structured light instead of laser. It uses the same type of camera, plain board used for only calibration and a projector which project the structured light on object. Since it use structured light instead for laser it only takes 2 to 4 sec to generate 3D model which is way improvement over just using laser but software is only available only for windows. There is still problem with making 360 models. It not possible to make 360 models when light and object are stationary. One of them has to move to make 360 models. Since prize of this kit is almost expensive as other professional scanner. Not many people buy this. People who get this is only to learn, or understanding how its work. David scanners don't have restriction on it; which make ideal choice on way to learn how 3D scanner work for beginners, and hobbyist.

3.3 Motor Research

3.3.1Non-feedback controller

Numerous of motors exist but question of efficiency plays a critical role. Stepper motors, Servo motors and DC motors with encoders are in general abundance on the current market worldwide and the desired choice of selection could be considered from some of their summarized guideline outlined principles below.

3.3.2 Servomotor

These motors usually operate by a known closed loop mechanism whereby feedback is integrated to a generalized position which may an angular or mechanical at a given point range .There are three types of these motors. These are the rotary servo motors are, AC Servo Motor, Brush DC Servo Motor, and Brushless DC Servo Motor .A servomotor which is a DC motor, are usually designed specifically to be used in a closed-loop control system. These motors are usually made of 3 major parts, a motor, a control board, and potentiometer which will be a variable resistor connected to the output shaft. The motor will be utilizing a set of gears to rotate the potentiometer and the output shaft at the same time. The potentiometer, which functions by controlling the angle of the servo motor, allows for more control of the circuitry to be able to monitor the current relative angle of the motor. The motor by means of a series of regulated gears, will be able to provide a turning mechanism at the output shaft and the potentiometer at the same time. The potentiometer will be fed into the servo control circuit and when the control circuit detects that the position is correct, it stops the servo motor.

These motors have much greater speed on average and acceleration as well. They usually have a power, ground and signal wires. Most designed motors have a red wire indicating the power, the black or brown indicating ground and the yellow, orange indicating signal. For appropriate position determination, the motor is usually designed in such a way that they have an additional sensor attached to it. The servomotor is going to be controlled by a microcontroller using a PWM wave generator which has a limited voltage and a limited current drive capability as well. To manage the needed servomotor's current and voltage requirement's, an amplifier will be in cooperated. Higher voltage ratings will be required to be able to make a rotation of the servomotor at efficient higher speeds and sufficient current levels are required to provide the torque or force needed to move heavier loads. This power is supplied to the servo control amplifier from the power supply in our case it will be from a main power outlet. As power is applied onto the servo motor, the load begins to move and speed or position changes accordingly.

The speed from the motor's shaft or the angle of rotation will be determined by a resolver or encoder. This will make an indication which will then provide a signal that will be sent back to the controller. The feedback generated will be a form of signal and will inform the controller's positioning and will indicate whether the motor is performing right or not.

The positioning controller looks at this feedback signal and determines if the load is being moved properly by the servo motor; and, if not, then the controller makes appropriate corrections. Greater occurrence with accurate angular position can be achieved with Servo motors

A well designed servo motor are usually made keeping in mind the relative maneuvering and the motors ability to being able to hold a given position between the desired rotational movements to be acquired. These angles may vary from zero degrees to 360 degrees. For global industrial usages, servo motors can be generally employed or designed in providing an appropriate actuation for numerous mechanics usages and set ups such as the common sewing machines, photo type setting machines and even in numerous factory machinery and automated mechanization. Below will be a typical servo motor specification

watts	100	Rated Output
RPM	5000	Max Speed
ib -in	3.124	Continuous Stall Torque
ib-in	9.116	Peak stall Torque
A	10.1	Peak Current
ib-in/A	1.168	Torque Constant
V/KRPM	13.8	Voltage Constant
ib-in^2	0.044423	Rotor Moment of Inertia
ms	0.96	Electrical Time Constant

ms	1.8	Mechanical Time Constant
ohms	2.4	Phase to Phase resistance
mH	2.3	Inductance
lbs	1.63	Motor Weight
mm	8	Shaft size
C	44	Motor Rated temp
lbs	0.66	Brake Weight
ib-in^2	0.00991	Brake Moment of Inertial

Table 3.3.2(1) Servo Motor Specification Table

3.3.3 HiTec Servo Motor

This servo motor product from HiTec HS brand, this has a digital metal gear servo although this might change since one of our aims to make an affordable scanner. This good product from HiTec is pricier. This has a relatively newer digitized model with higher programmable performance features including a recommendable resolution, greater center holding torque with a newer included metal gears, the old featuring top ball bearing and lighter weight compared to the other brands. This has an unbelievable quicker transmit time of 0.10 seconds in as little at 6v. This might be a little more than needed for our design but we will love to work with it.

24.99/30.55oz	Torque weight
4.8v/6v	Voltage requirement
23.60 * 11.60 * 24.00mm	Size
Ball Bearing	Type
0.14/0.11 seconds	Average speed of motor
60 degrees	Operational Angle
0.10seconds	Transmit time

Table 3.3.3(1) HiTec Servo Motor Specification Table

A radio form of servos exist which could be radio controlled if desired exist in the market and have various advantages. An advantage of this radio control is the ability to be able to mechanically link to a potentiometer. This could serve as a huge advantage. By means of a standard RC, a pulse width modulated signal is received by the servo through a standardized signal wire. Circuitry configuration designed in a given servo has the ability to be able to disseminate the width of a given pulse into a relative position. When a command is giving to the servo to begin a rotation for analysis, the servo will be powered until the desired value of the command relative position is reached on the potentiometer.

Normally, a given servo motor pulse value of 1.5 ms width should be able to set the servo to its "neutral" relative position or a value 90 degrees. By the same concept, the value of a pulse of about 1.25 ms should be able generate its setting value to 0° and finally a pulse of 1.75 ms should set the servo to 180 degrees. Obviously all these values for the timing values of the servo should vary among models on the market. But generally, the neutral relative position should be around 1.5ms

3.3.4 DC Motors

DC motors are generally helpful in designs which might be needed for a greater amount of speed movements and torque requirements. The motor usually has an encoder which basically functions as a converter and provides the relative position of the motor of the shaft into a code. The attractive and repulsive properties are the basic working principle behind any magnets. Noncomplex motors have electromagnets on the shaft which helps in the attractive and repulsive mechanisms. The electromagnets have the ability to flip their magnetic field hence their basic usage. DC voltages have the ability to provide a greater impedance if required DC motor are defined in terms of motor speed, and they are divided into two cases. In the first case, the motor operates above base speed with the switch closed, by-passing the field winding to the armature (that is, Rp< 1)

Generally servos offer better and adequate power and would be more compact than other available motors on the market. An advantage of using these servos comes with the ability of being able to control or in regulating to our desired speed and making it more ideal for open loop systems applications .One of our general objectives will be focused on the effective communication between the microcontroller, laser unit and the motors we are about to use. A stepper motor might be considered for the laser unit. Torque speed factor will be considered for the rotational platform. We are considering using a servo for our rotational platform as well

The motor will be able to run on a 1.5-3VDC. At 3 volts with no load and would definitely be needed to step down in order to be operational on our available power supply rail which is running approximately 12 volts. The motors speed is around 14200 RPM which is desirable. The motor when operational draws a current of about 300mA. It is rated at maximum efficiency output of about sixty percent. The gear motor has a functional encoder built in. The motor being considered is meant to be used at 12V, but the motor has the capability of rotating at lower voltages. It can be even rotate at 1V. There are six face plate mounting holes which are spaced evenly around the outer edge. The mounting holes then forms a hexagonal shape and there are 15.5mm apart hole at the neighboring centers. There are location feedback from these motors.

5500rpm[3]	Free - run speed
250mA	Free run current at 6V
2500mA	Stall current at 12V
5 oz. in	Stall torque at 12V
11 in	Lead length
4mm	Face thickness
20 degrees	pressure angle

Table 3.3.4(1) Location Feedback Table

3.3.5 Stepper Motors

Stepper motors exist as being electromechanical actuators that has the capability of converting its input digital to an analog motion. Stepper motors are generally referred to as step motors and are usually brushless Direct Current electric motors which has the ability in dividing a full rotational movement into an even or number of equally desired steps. This is made to happen through the motor's controller electronics. Several stepper motors have been developed over time and examples include the solenoid activated, the variable reluctance, permanent magnet and synchronous inductor. Independent of stepper type, all are devices that index in fixed angular increments when energized in a programmed manner. The step motor's relative position achieved could therefore be giving the desired command to make a movement and be on hold at one of those desired steps without the possibility of any feedback sensor or commonly known as an open loop controller keeping in mind the motor should be well sized to the given application. Most stepper motors are conveniently used in an open loop system configuration, which normally could generate oscillations. To avoid this from happening, a complex circuits or a feedback mechanism is used, this resulting in a closed loop system. A stepper motor will be more extremely useful in available applications where the controller signals will be appearing as pulse trains. One pulse will cause the motor to increment one angle of motion. This will be repeated for at least one pulse. Stepper motors' normal operation is usually made up of discrete angular motions with a uniform magnitude with respect to a continuous motion. Stepper motors in general have a limited horsepower to about one horsepower and approximately 2000 rpm. This serves as a main disadvantage to be used in our design.

Available are the abundantly known switched reluctance motors which is mechanically designed in such a way that power is not needed to be delivered to the available movable parts being concurrently used. Although these are very useful in the industrial market due to its advantageous property of having large stepping motors which includes a reduction in pole count, and widely considered as a closed-loop commutated. With modern electronic devices this is not a problem, and the SRM is a popular design for modern motors. Its main drawback is torque ripple.

For the sake of this project we are considering careful analysis of all kinds of motors before we conclude on which kind of motor we decided to pick. Various affordable stepper motors are designed for maximum torque efficiency and some even operate at lower speeds. One of the factors we will be considering will be how much power consumption the stepper might need since we want all total power consumption to be around 35 to 36 watts

The widely known stepper motor when carefully designed has the ability in providing an uninterrupted revolving movement, ability in providing a precise locational analysis control with the controlled feedback mechanisms not being a subject of interest. Since steppers mainly move about their numerous poles, the extra encoded which are mostly required in many motors are not required for their operation. This could be a very critical advantage considering cost as a major factor in order for the project to consider a success.

These non-encoder required motors are often required in a more useful generalized open loop systematic applications. Outer rims of motors normally operate by electromagnetic mechanism.

The whole entire mechanism are powered enough to be able to provide the needed designed motion that is required from the motor when functioning. Due to variations required for the motor, the stepper could be outlined and programmed to achieve various angle variations since these are variations required of the motor. However a disadvantageous scenario could arise where there is an observable amount of heat from the operating motor and the drive as well.

The hybrid bipolar has a varying angle of 1.8 degrees and has a 200 step per revolution. It has a good current consumption and relatively good voltage.
The motor consist of black, red, blue connecting wires controlled by bridges. The stepper motor will not be ideal for our project since it does not provide a greater torque

The way a stepper motor works is that it has coils wrapped around themselves on every side of the magnetic turning shaft (usually 4 sides), and inducing current in each one of these coils, in a series order, makes the shaft turn by one step every time we switch which coil the current is delivered to. Even though the functioning of the motor is very analogue, continuous and heavily relying on induction, the overall behavior seems pretty discrete as the motors are mechanically designed to turn on a pre-specified step angle. The poles of the magnetic field are actually made of teeth, one set on each coil or field source, and on the main rotor in which the shaft is attached. When current is applied to a coil and magnetic field is created, the two teeth set geared up together aligns itself and the rotor turns.

The size of these two elements and the number of teeth on the rotor and the stator is actually what mechanically sets up the rotating angle, the field-rotation being well defined. Although providing current to the coil is viewed as a “pulse”, it’s only its finality that seems behaving that way. In fact, if we continue applying current to the same coil, nothing will happen as the magnetic field will remain stable, with no pole inversion and no induced movement of the shaft-stator system. The pulse then only models the period of time in which current application is actually effective, just enough time for the mechanical system to come to an end after the pole inversion rotates the shaft in the desired way. This is why a stepper motor requires a constant toggling power way to keep working.

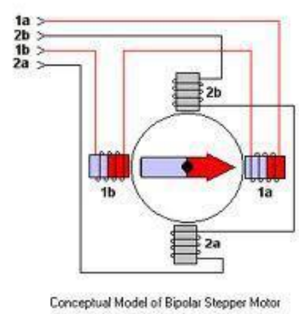


Figure 3.3.5(1) Conceptual Model of a Stepper Motor (Pending Permission)

In a bipolar stepper motor, there are then 4 power lines supplied to the different sides of the motor, here denoted 1a 1b, 2a and 2b. This is what we are going to control as we power up or power down each coil a different times. Bipolar stepper motors, although a little more complicated when it comes to operation and internal circuitry, usually provide more torque for the application, which is an interesting advantage.

NEMA-17 17HS15-0404S - After brief comparisons of different stepper motors that seem to meet our requirements, we start looking at the NEMA-17 17HS15-0404S bipolar motor that has a 1.8 degree step among other characteristics that interests us. Other characteristics are displayed below:

Value	Characteristic
12 Volts	Voltage
0.4 Amps	Current/phase
2	Number of phases
56.7 Oz.in	Torque
36 Ohms	Coil resistance
$ 9.94	Price

Table 3.3.5(1) Specifications of the NEMA-17

When comparing the holding torque of this version of the NEMA 17 to other more powerful motors, we realize that this is a pretty low value knowing that we want to maximize how heavy of an object we want to be rotating. While we wait to get an idea of what the holding torque physically represents, or give a meaning to this so-far arbitrary number, we will examine another, more powerful type of motor to maybe elaborate an alternative design in case this motor is insufficient.

NEMA 23-57BYGH310 - While it has a lower nominal operative voltage, it has a much higher current per phase rating, from which comes a nominal 16.6 Watt of power and the phenomenal torque. We will continue the design with this second choice of motor in mind until we have a better idea of the real-time torque requirement. The price of stepper motors goes up as their power increases. In fact, for NEMA 23 types of motors and the models above that, offering more and more torque, the price goes up in increments, as we go from hobby oriented, low torque stepper motors, to more capable, industrial grade motors. As we need to minimize the costs as much as we can for the validity of our project, the cost of the stepper motor is definitely something to keep in consideration because that might make the final bill go up in an undesirable way. We can see here see that the holding torque produced by the NEMA 23 is way higher that of the NEMA 17, and will definitely meet all weight requirements. We can take the example of this NEMA 23, bipolar motor as well, in which the concerned specifications are:

Value	Characterisitic
8.3Volts	Voltage
2.0 Amps	Current/phase
2	Number of phases
333.1 Oz.In	Torque
4.3 Ohms	Coil Resistance

$23.89	Price

Table 3.3.5(2) Specifications of the NEMA-23

3.4 Motor Control

3.4.1 Control Methods

Control through FPGA - An option to consider for controlling our motor is a Field Programmable gate array (FPGA). Advantages of FPGA include, as the name mentions it, the flexibility and recycling abilities of the FPGA. This gives us a lot of room for re-experimentation and error margin during prototyping. Programming an FPGA is also relatively easy for our purposes; since you can use software such as Xilinx ISE and upload the design to the FPGA using whatever interface goes with the manufacturer of the device. We can also use a hardware description language in parallel with the graphic programming interface. As simple one is VERILOG, which has lots of characteristics of a basic high-level such as C.

FPGAs are known for their high sense of reliability, trusted by a lot of companies that make risky products, notably in the defense and aerospace industry. We can use reliability in our project as the process of functioning is time sensitive and every step is very crucial for the whole duration. Besides Xilinx's Basis boards, Altera, one of the leading top-notch FPGA manufacturer out there, offers some pretty adequate boards very suited to control motors in general, including DC motors and steppers motors. Running an FPGA on a costume PCB will also I have us the opportunity to run feedback control system to have more monitoring over the behavior of our motor. However, we don't find this that much of an advantage for our needs. When it comes to the kind of motors that interests us, feedback option is usually a characteristic for stepper motor like we saw previously. FPGAs permit us to by pass this deficiency for stepper motors by giving us feedback while keeping the many advantages: precise hardware function, and punctual, well defined holding torque at each degree rotation.

Not only we can visually see the action of our motor-disk platform, but the way the pictures are going to be taken totally depends on the position of the motor, as in every snap will only be taken if the position to the next degree of rotation has been reached. This being said, we shall not reject the possibility of implementing this option, because even though not vital it would be an addition that could add some more reliability in our system, as well as a more modern, professional approach to controls.

Microcontrollers - Here for our purpose, using a microcontroller can be an easy and reliable way to control our spinning motor. The fact that we can reprogram it with ease will make prototyping and testing easier task once the design phase is over. For example, we may need to change the amount of degrees per rotation of the disc that goes with each capture from the camera. This can be both achieved with servo motors and stepper motors. One would ask itself why would we, for as simple as a job as rotating a motor literally opt for a tiny computer to do the job?

The answer is the state of technology. Low end microcontrollers have been democratized both in the professional engineering world as well as the hobby world, which in consequent has made them more accessible, cheaper, and with more resources and information about them for an optimal use. Reliability is an important factor too, because it is in fact the point that is criticized the most about such digital systems, in opposition to the analog alternatives and automated circuits.

In the market today, we denote two main families of microcontrollers that would fit out needs: The PIC microcontrollers made by Microchip and AVR microcontrollers manufactured by Atmel. In the big picture, both processors use a RISC (Reduced Instruction Set Computer) which is simpler than CISC (Complex Instruction set computer) that we find in most personal computers nowadays, with the most famous microprocessor manufacturer, Intel, equipping the world with these devices.

That leaves RISC for the majority of the other embedded application out there, just like our needs. The only downside of RISC is that since the architecture is simpler there are fewer instructions to choose from to perform our functions, and in consequent the embedded programmer has to use more instructions to do the same job, using more time and power. It is however something not to worry about since we do not have a complicated computer program to run. AVR and PIC are both 8-bit architecture devices, the simplest form of microprocessors that are widely available.

AVR Microcontrollers - AVR is based on what is known as a Modified Harvard architecture, in which the instructions are stored as data making the processing more flawless. They are 8, 20 or 40-pin devices, very popular nowadays for the single reason that development board manufacturers use them extensively; they are very practical to integrate with other components making a more complete embedded system. One of those development boards is Arduino. Even though there is a wide range or Arduinos out there, they all use a suitable ATtiny or Atmega microprocessor depending on the specifications and requirements, as well as what peripheral components of the board. The ease of use makes it a to-go choice for hobbyists to control their inventions, opening up endless possibilities both towards hardware purposes or straight software. One particularity of the AVR is that it is very C friendly. These processors were intended to handle compiled C code, their creator took in mind the way C code is written and designed a pipeline system very suited to that type of logic (making it among the fastest 8-bit CPUs in the market), with a very adequate assembler. The Assembler is available for free on Atmel's website and is called AVRtools, part of Atmel Studio. In addition, Atmel studio offers other software facilities to help the beginner or the more experimented programmer with additional options. Although, we may consider using a third party software to program our microcontroller.

Our requirements being the most simple, we will look at the lowest grade processor; the AtTiny. After some research on the current market, an AtTiny 25/45/85 with a starter development board to would be a way to go with these, and if this is what we are basing our choice on, AVR is the natural choice. An AVR ATtiny chip costs between $2 and $3 in most online stores.

On Atmel's website, an original development board made by the manufacturer goes for about $80 (refereeing to the SPK500, entry level programming/development board designed not only of Attiny but also the more complex, higher pin-count devices). Other programming interfaces include the following Development board and programming interfaces:

Price	Link	Name
$19	https://www.sparkfun.com/products/11801	Tiny AVR programmer
$10.45	http://www.mouser.com/ProductDetail/Olimex-Ltd/OLIMEXINO-85-ASM	Olimexino 85
$14	http://www.karlssonrobotics.com/cart/pocket-avr-programmer/	Pocket AVR Programmer

Table 3.4.1(1) Comparison of AVR IDE prices.

Key Parameters

Parameter	Value
Flash (Kbytes):	8 Kbytes
Pin Count:	8
Max. Operating Freq. (MHz):	20 MHz
CPU:	8-bit AVR
# of Touch Channels:	3
Hardware QTouch Acquisition:	No
Max I/O Pins:	6
Ext Interrupts:	6

Figure 3.4.1(1) Overview of ATtiny45 MCU (Credit to Atmel.com)

PIC Microcontrollers - PIC is AVR’s main competitor in the market, and hence shares lots of similarities between the two architectures and specifications. Yet they differ in a few different ways. They are from a modified Harvard architecture as well, But the pipelining system is quite different and reputably less efficient, but simpler overall. There is one or two instruction per cycle, and the processor is designed with the best speed to cost ratio in mind. When it comes to programming, lots of development has been made by Microchip as well to facilitate the process for the majority of users. When it comes to our requirements, the PIC would meet them as well because of its similarities with the AVR.

The input voltage is between 2V and 5.5V. The maximum voltage is the same for both microcontrollers, so we are going to assume that our power supply, which is discussed in the power section of this document, will be of an optimal 5 Volts for steady results. The only time we would need a different power intake would be during the programming of these microcontrollers (High voltage programming versus low voltage programming, as described in each chip documentation by Microchip or Atmel). A great advantage of the PIC over its rival is that it can be continuously programmed independently of the position of the RESET pin. Whether it’s LOW or HIGH, the RESET pin will not affect the programming process when hooked up to a serial port.

This will be particularly useful if we opt for In-circuit Serial Programming (ICSP) and we use the RESET as one of our Inputs/Outputs. We predict a relatively minimalist design for our project so that might be a strong possibility. The price of a PIC12 is typically, as found on various internet stores, between $1 and $2. The high production industry of these chips makes them very cheap devices, whether bought in bulk or not. Here are a few programming interfaces for the PIC available for purchase, to give us a general idea of the other thing to buy in complement.

Specification	Value
Family	PIC12F675
Number of Timers	2
On-Chip ADC	4-chx10-bit
Program Memory Size	1.75KB
Program Memory Type	Flash
RAM Size	64Byte
Interface Type	RS-232/USB
Family Name	PIC12
Minimum Operating Supply Voltage	2V
Typical Operating Supply Voltage	2.5V|3.3V|5V
Maximum Operating Supply Voltage	5.5V
Minimum Operating Temperature	-40°C
Maximum Operating Temperature	85°C
Screening Level	Industrial
Mounting	Through Hole

PIC12F675-I/P: PIC CMOS Microcontroller DIP-8

Specification	Value
Device Core	PIC
Packaging	Tube
Pin_Count	8
Supplier_Package	PDIP
Product Width	6.35mm
Product Height	3.3mm
Product Length	9.27mm
Product Type	Microcontroller
Data Bus Width	8Bit
Instruction Set Architecture	RISC
Maximum Clock Rate	20MHz
Maximum Expanded Memory Size	14KB
Maximum Speed	20MHz
Number of Programmable I/Os	6

Figure 3.4.1(2) Overview of PIC 8-pin microcontroller (Credit to Microchip.com)

Price	Link	Name
$44.95	http://www.newark.com/microchip/pg164130/debugger-pickit-3-icd-pic-dspic/dp/25R8311?CMP=KNC-GPLA	Pick It 3
$14.50	http://www.ecrater.com/p/16756443/pic-usb-automatic-programming-microcontroller?gps=1&id=52211034499	PIC Programmerk150

. **Table 3.4.1(1) Comparison of PIC IDE prices.**

When comparing both characteristics of out two microcontrollers of choice, we quickly realize that they are both very similar in the details. Two important factors are the maximum overall clock rate, both rated at 20 MHZ, and the number of input/output pins. Both have 6 that we can use, even though we can most likely limit the pins used while programming, given that we need two fundamental action controls in our design (one pin to turn the motor left and another one for the left functioning of the driver).

We still then have to make a choice, and after realizing that these two micro control units are pretty similar in terms of specifications, one shall look at other parameters. Next comes the cost. At our level, we will only need one processor, but we may want to order a few more in case they get damaged during the testing and prototyping phases. 5 processors would be plenty.

AVR ATtiny45	PIC12	Number of devices
$2-3	$1-2	1

$10-15	$5-10	5

Table 3.4.1(3) MCU Prices

As we can see in the previous table, the total order would be way cheaper if we chose the PIC route. However, 10 dollars is only a fraction of our total budget and should not be our most important criteria.

Our next and final decision point will come from what will make us achieve the best design, or the most effective/reliable. From the different development boards that we have reviewed, we can see that the more complex the board is, the more expensive it is, especially when it is officially made by the manufacturer of the chip. From an engineering point of view, the simpler the design is the most efficient, and the most likely to be reliable. So avoiding the development board route would not only minimize the cost of the final product, but would lead us to a simpler, straight forward design. Designing our micro control unit right into the final printed circuit board would then be our solution.

That means we will have the option to program the chip while already in place in the final product while being hooked to the motor system. The way to do that is to design an ICSP (in-circuit serial programming) on the PCB, and it will be as simple as driving the concerned pins out of a header by themselves, ready to be connected to serial “programmer”, which is a cable driving those pins to a serial port in a computer. Then we would just need to transfer our program file (most of the time it’s a .hex file, for both MCU families) into our board, according to the specifications of each manufacturer. In fact, both Atmel and Microchip have ICSP documentation available on their website, and when joined with the information about each particular microcontroller, gives us enough tools to program the MCU into the target board.

The decision then will come from the final step, the actual programming. As mentioned before, AVR’s architecture is way more modern and friendlier to program, with a wide availability of resources and compilers, including C compilers, which we will first try to use with a classic programming approach as opposed to assembly programming. For that, we will start our design with the Atmel AVR Attiny45 microcontroller that we previously researched.

3.4.2 Motor Driver

Now that we figured out the ins and outs of controlling the motor, we shall think of how to power it properly. It is sure that the input signal from the microcontroller will deliver power to the motor, but being a control signal out of the MCU, it will have too low voltage and current to properly drive our stepper motor, which is going to be from a grand maximum of 5.5 Volts. This is too low for the characteristic voltages that we see from the research on motor, which varied from higher values in order of 12v to 36v.

They were motors that used required less power, most of them including 5v, the voltage of the MCU, however these were often synonymous of lower transmitted current which lead to having a lower torque, characteristic that we should not neglect for the well-functioning of our system (see table). Instead of trying to capture 5V on one side, and 12 volts on the other, out of the same power source in our motor, we are going to plan on having two different supply routes for the two different voltages, in order to simplify our circuitry when to goal is reliability of control of the motor and a full power delivery so we don't lose any current and torque in the process. So we are looking at having two independent power incomes inside of our board, one going straight into powering our motor.
The obvious solution here is power up our motor separately from the control unit, and for that we need what we call a driver. We will evaluate the different options we have, between designing it from electrical components from scratch, and using preexisting circuits or ICs to integrate in our final printed circuit board.
The first option we found to drive our motor is to use an "Eight Darlington array" IC module, which consists of 8 bipolar junction transistor setups, most of the time as NPN, or more precisely they are called Darlington transistors. This array, pretty simple overall, allows for a very high voltage output in its branches, while maintaining the switching between the different inputs/outputs pretty fast, for an accurate digital controlling and real-time application just as we need.

Coupled with each one of these transistors in the arrays, is a fly back diode that eliminate the voltage spike or irregularities in the signal when, just like our application, the voltage is suddenly cut between two common cathode linked to each other, making the transitions between all the pins the smoothest possible. A family of these integrated circuits is the ULN2801A, ULN2802A, ULN2803A ULN2804A and also ULN2805A. Offering different voltage configurations.

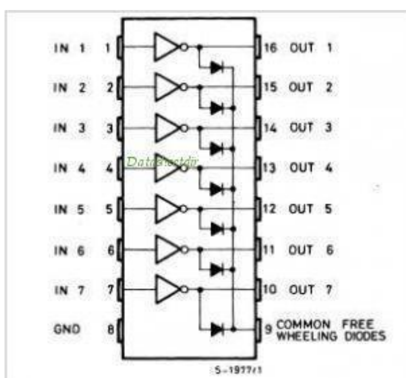


Figure 3.4.2(1) PINout diagram of the ULN2804A (Approval from T.I.)

The big problem we see with these devices is the maximum output current it provides: 500 mA. This value would be adequate to the lower tier of motors in term of power, and sub consequently of holding torque. Also being operated at 12 volts it is indeed enough to drive and power up the NEMA 17 type of motor that we saw, especially when the current per phase in this one is only 350 mA. But in case this stepper motor is not enough, we shall look at some more powerful driver chips to drive potentially bigger motors.

The next integrated circuit motor drivers we are going to look at are all adequate for bipolar stepper motor driving, and provide levels of performance superior to the simple Darlington array. These are the T.I SN754410, The Allegro A3977, The ST L298 and also the T.I DRV 8825. After using the datasheets provided by their respective semiconductor manufacturers, we took a few specifications that are the most important to us and made a table to compare all those.

DRV8825	**L298**	**A3977**	**SN754410**	**SPEC MODEL**
8.2V to 45V	7.8V to 46V	8V to 35V	4.5 V to 36V	**Operating voltage**
2.5A	4.0A	2.5A	1.0A	**Max output current per phase**
STEP	Classic 4-wire	STEP	Classic 4-wire	**Control input type**
28	15	28	16	**Number of pins**
$22.00 for 5	$2.84	$4.00	$2.57	**Price**

Table 3.4.2(1) Motor Drivers IC.

The “control input type” refers on how the microcontroller is going to provide the information to the driver. A classic 4-wire way is how we would do it with the ULN200xA driver, which means that the controls are purely driven through the transistors, so that the inputs are equivalent to the outputs, with the difference that the outputs will have all the electrical proprieties to properly supply the motor coils (ref. ULN diagram).

The next thing to look for is the output current per phase, because that is what will ultimately determine how strong our motor can be. All the drivers have at least 1A of output voltage which is plenty to power up stepper motors that are in our price range. Not only the SN754410 is the cheapest, but it’s also of the simplest kind, with only 16 outputs and inputs, this will allow for a more straight forward design without useless features, our goal being simple and the way to control it well defined. It’s rated at 1amp, and from our research on motors, we can definitely select a model fitting this criteria that will allow us to maximize the torque delivered, and minimize the cost of the actual motor.

As we can see in the image below, other drivers re way too complicated for the simple purpose of our choice of function. Hence why we are going with the simpler one.

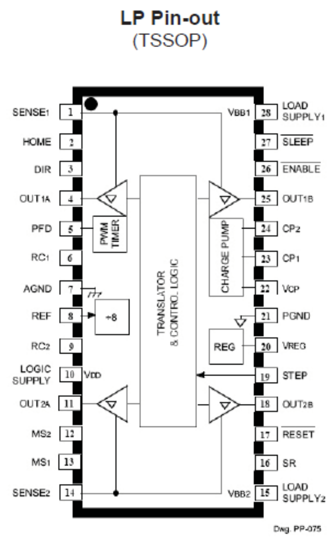


Figure 3.4.2(2) A3977
(Permission from allegro)

3.5 Laser vs. Structured light

Nowadays more and more people using non-contact 3D scanner over contact base scanner, due to usefulness technology and how it works. There is lot of product use this technology for scanning object, scanning environment, accuracy, resolution. But mostly noncontact base scanner either use laser or structured light. Both types are optical light based measurement techniques. And both are good for different applications. Structured light is more accurate in general, but with laser light scanner you need to use multiple laser and scan it from multiple angle to get same accuracy but it’s still cheap and have big range. Whereas structured light projector cost more than laser base scanner.

3.5.1 Laser

Laser scanners use laser light to scan parts. 3D laser scanners commonly found today are mostly either: handheld where laser line sweeps over the object or, tripod mounted where the laser line sweeps across a field to cover area. All handheld type is either attached to an arm or has markers on the surrounding to give constant measurement of where the scanner is at during the scanning process or it can have sensors chip to tell position of the scanner. Other way to align scanner is to put marker on the parts as a basis for aligning scans to complete the 3D model.

3D photography using planer shadows where camera and the light source must be separated so that it cast shadow and camera rays should meet at angle which is easy to read. The stick is slowly waved in front of the point light to cast a planer shadow that translates from left to right in the scene. The position and orientation of the shadow plane, in the world coordinate system, are estimated by observing its position on the planer surfaces. Before that it also required calibration of camera. Calibration of camera is necessary to transfer image measurements into the world coordinate system.

3D model can be recovered by triangulation of each ray by the shadow plane. To reconstruct a 3D model, we must know the equation of each shadow plane in the system coordinate. The cast shadow will create four distinct lines in the camera image, consisting of pair of lines on both the horizontal and vertical calibration boards. These lines represent the intersection of the 3d shadow planes with the calibration boards. Where we need to estimate the 2D shadow line projected on the horizontal and vertical planer regions. To perform this, similar approach is used that explained by Zhang method. Then we have to evaluate the maximum and minimum brightness observed at each camera pixel over the stick-waving sequence. To detect the shadow boundaries, choose a per-pixel detection threshold which is the midpoint of the range observed in each pixel.

3.5.2 Structured light

Structured light scanners are mostly mounted to ceiling or ground, where a stripe pattern is generated by scanner’s projector and is laid over an area over the object. Overtime, the stripe pattern is modified in width and phase and 3D scanner abstract the 3D coordinates from calculating the returned patterned. This type of scanner also call white light 3D scanner because its use bulb to generate white light. Nowadays people are replacing to the blue light because blue light give accurate scan due to it wave length. We also need Structured light scanner to capture colors of the object which you can’t do that with laser scanner. The primary benefit of introducing the projector is to eliminate the mechanical motion required in laser light scanner, and it could be cost efficient as well. One key point to remember using structured light scanner is the camera and projector lenses determine a scanner’s field of view (FOV). FOV is affect by camera lens being used, and the lens on the projector, and distance between the scanner and the object. With same distance a lens with small focal length allow to see more of the scenes whereas lens with large focal length sees less but gives more detail. Also camera and projector lens focal point should match.

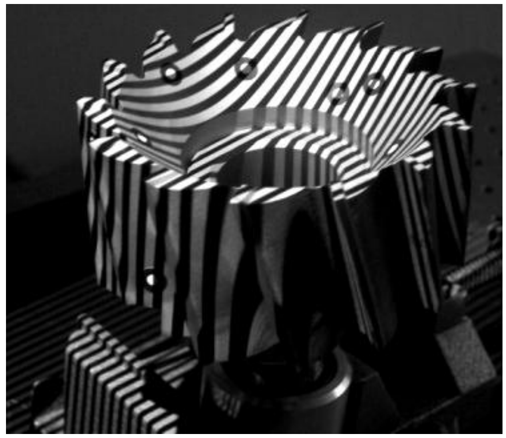


Figure 3.5.2 (1) Structured light
(Credit to spatialvision.com)

To calibrate assuming minimal lens distortion, the projector can be used to display a single column of white pixels transmit against a black background; images would be required to assign the correspondences for calibration. It would be between camera pixels and projector columns. After calibrating the system, a 3D point cloud is reconstructed using triangulation. This simple stripe lines sequence does not give good result. Since we are free to project any 24 bit color images, we can project color strip patterns which allow the projector-camera correspondences to be assigned in few frames. In general, the identify pattern with accuracy with single frame or multiple frames has benefit and downside. Purely spatial encodings allow a single static pattern to be used for reconstruction, enabling dynamic senses to be captured such as if object is on rotating stage while scanning. Alternately, poorly temporal encoding is more likely to benefit from redundancy, which reducing reconstruction. Calibration can be done using MatLab toolbox or OpenCV. And also the project and camera both need to be calibrated. To estimate the projector parameters, a static checkerboard is projected onto a diffuse planer pattern with a small number of printed fiducially located on its surface. Or printed fiducially were horizontally spared by some fix length and vertically with different length. The camera and projector calibration obtained using same procedure as checkerboard.

At a basic level, we are concerned with assigning an accurate projector column/row to camera pixel correspondence, otherwise triangulation will cause to large reconstruction errors. The decoded set of camera and projector matching point can be used to reconstruct a point cloud. Then the projector column matching point has to be reconstructed using the projected row point. Then the projector pixel to camera pixel corresponding point can be used to reconstruct the point cloud using ray triangulation by finding the closest point to the optical rays defined by the projector and camera pixels. Then a simple per point RGB color can be assigned by sampling the color of the all-white camera image for each 3D point to make it color copy.

Not all objects can be scan with 3D scanner there are some object that are very difficult or nearly impossible to scan. Such object has highly reflective surface or if it transparent, translucent or emissive. If the objects has highly occluded surfaces with no line of sight it's gone be nearly impossible to scan all the parts or if has hard edges and deep concave segments.

Now, let's look at pros and cons of both type of scanner. Laser scanner scan the object only one time as line laser passes the illuminated object. Whereas structured light scanner scans the object multiple times with stripe patterns varying width and phase. This will estimate the shape of the object. This gives batter of accuracy of the final result is estimated from multiple scan object therefore it's less chance to make error. But laser scanner has a speed advantage since they can do a single sample faster than the multiple sample scan. Now day with advanced processors, and cameras, this can capture the scan in less than one second with a million points. Due to fast scanning speed of structured light scanners, turnout useful in body scanning application where people have difficulties to stay still like mundane object. Laser scanners are generally based on a point that is split into a line, which is then swept across a plane. They are performing area scanning by stretching from a one dimensional line to two dimensional areas. Structured light scanners are true area scanners because of the projected patterns being on the two dimensional and cover the entire area. For the lighting conditions laser scanners have the ability to turn up the gain to get some data in difficult high lighting environments. But it's not always possible. For structured light scanners are limited by the light intensity of the bulb and contrast level of the projector. It's difficult to use the scanner in ambient lighting environments. Lasers also remind the word safety, due to their ability to focus light intensity and energy into very small beam of light. The biggest concern is eye-safety danger. Some of the laser used in 3D laser scanners today is are not the safest laser. But there are laser systems intended for human scanning such as face and body scanning. There scanners need to be rated for eye-safe and are certified to be below class 2M where laser has larger diameter mean low intensity. Structured scanner being based on simple white light, don't have that problem and are much less concern for users. Other structured light scanners, such as blue light scanning, are also not a concern for safety since they are not dealing with the intense light.

3.6 Calibration

Three-dimensional scanners measure shapes by acquiring a set of points, (cloud of point), which is a discrete approximation of a real object. The dimensions of the digital representation should be as close as possible to the dimension of real object. To be accurate, measurement of real time object should match the measurement of 3D model or at least should represent on scale. That's how accuracy of 3D scanner is measure. Accuracy defined as "the maximum radius of the space inside which there is total probability of finding the coordinate of a measure point." From this we can conclude that smaller this radius is, the better the approximation of the reconstruction to the real object will be. To make it accurate mode camera should be calibrated to get accurate measurement.

The Projective calibration of cameras rely on the identification of good set 2D to 3D point correlation,
This has to do with parameters like focal length, principal point, skew coefficient, distortion and others. Some other parameters can affect like rotation and translation between the camera coordinate system and object. There are few well defined methods available such as Heikkila&Silven or Zhang Methods to overcome problem.

Zhang's method uses a checkerboard pattern, which is placed in front of the camera. And said at least three different images in various angles and position must be acquired for the computation of camera calibration. Then program use for detecting corners of the checkerboard pattern. Then these points are used in calculations to figure out measurement. This method is also used in OpenCV. Downside to this method is 2D calibration pattern usually takes long time. A better solution would be to use 3D calibration object. Where object could be placed in the scanning area and the calibration will be carried out during the scanning process.

This method the calibration need necessary to adjust the 3D scanning system. This should cover the whole area of scanning space (world). This means determining the position of the camera, zoom and focus on the scanning surface. After the calibration it is no good idea to manipulate these settings after calibrated. Or we have to calibrate again. It is necessary to use appropriate calibration objects, or shapes. These patterns are than used in the calculations of intrinsic and extrinsic parameters of the system based on images. The most commonly used is the checkerboard pattern, which is provided in some well know program for image calibration like MATLAB and OpenCV. This pattern is printed on flat surface and has predefined dimensions and parameters like its 20mm by 20mm each box. There is also many other similar pattern types that can be used. Most of them are planner based but some are also represented in 3D space. Simple method like Direct Calibration based on a simple idea. If the camera coordinates of a sufficiently dense grid of workspace points can be measured accurately, then the position of any point in the workspace can be obtained by inverting the resulting world to camera maps. A calibration need to be done on each camera in system. In order to detect calibration points in the Y direction, the block is formed by 20 parallel slices spaced regularly.

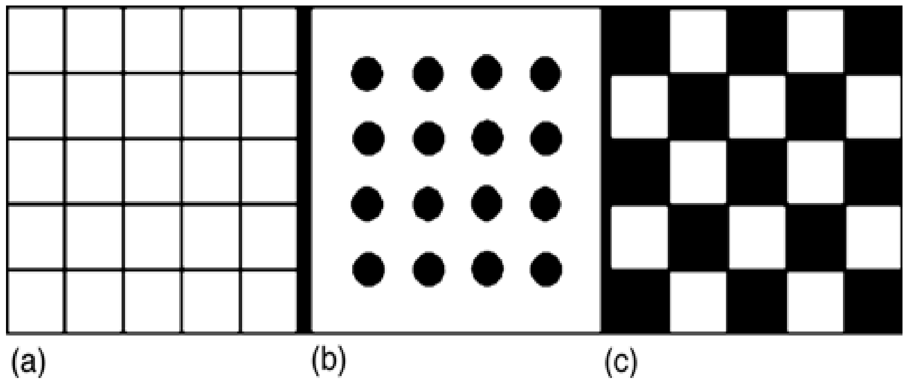


Figure 3.6.1(1) Calibration pattern
(Reference from spiedigitallibrary.org)

Since we are planning to use MatLab and OpenCv for our project, let's look at calibration toolbox for MatLab. The toolbox provided by MatLab is really good tool when it comes to mathematical computation. Which also provide us tool to calibrate the cameras of the optical based 3D scanning system. The Advantage is the use of open source code, which is use for implementation of its functionalities in various three dimensional scanning system like projector-camera or camera- camera based systems. It's a fundamental building block that can be extended to cover the overall calibration of the 3d scanning. It provides many tools to analyze the results of calibration; Toolbox also provides tool to visualize the results of the parameters which is necessary for calculation of distances, and tools to visualize the distortion of camera and we can see the distribution of projection errors. Some of the other good things about this is since tool is programmable it is possible to make change and implement your own features if you have to add more camera or projector.

Open Computer vision (OpenCV) is a set of open source libraries for application in vision systems. There is lot of optimized algorithms available on OpenCV in many different programming language and can support many operating system. Since these libraries are open source, which make easy to edit the file according to your requirement. Unlike camera calibration toolbox for Matlab this calibration method is fully automated and can be implemented also in real-time computer vision system. It contains algorithms for automatic detection of the calibration pattern with the detection of inner intersection points of the checkerboard. You still need to know type of pattern, size and distance of the objects in the pattern to calibrate perfectly.

3.7 Data Capture

3.7.1 Laser Triangulation

Triangulation method recovers the scene depth. It’s called triangulation because a laser source, camera and object form a triangle. It uses geometric principles of triangle to measure the dimensions of an object. In general procedure has two main steps. One identifying the illumined pixels by laser, we can use filter: where we subtract one image from another; one taken with laser off and one taken with laser on. And second step we use equation (trigonometry) to calculate the distance. This method is only useful for short range scanner. Because it’s hard to capture one laser line that more than five meters away with average camera.

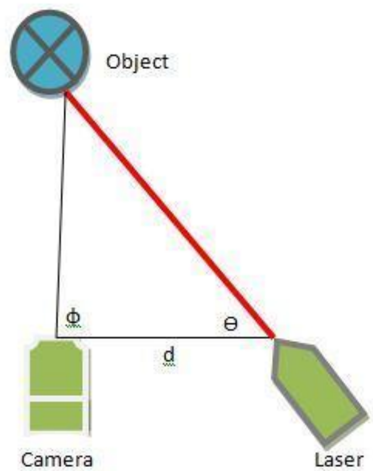


Figure 3.7.1 (1) laser triangulation
(Reference from irawiki/optical-triangulating)

Triangulation method is used to locate the position of an object in relation to two fixed points. For our case two fixed points going to be camera and the laser line, which then form a triangle. And the distance between the camera and the laser ‘d’ has to know, and the angle of the laser corner ‘Θ’ is known. Then angle of the camera corner ‘φ’ can be determined by looking at the location of the laser line in the cameras field of view. This allows to determining the shape and size of the triangle and respectively the 3D position of the laser line.

Every point on 3D model is calculated with its unique triangle; we need to have good processor to do calculation for good quality scan. And it’s not possible to write codes that allow for real-time scanning, this method do all the calculation in series. To do real time scanning different method can be used like stereo vision or structured light. The idea of calculating distance from triangle comes from pinhole model which we explain above, which is the simplest geometric model. This is basic of triangulation technique turn out to be accurate but not compare to some of the others scanner because this method only use one beam of light which turn out to be cost effective, but real world scanner uses numerous points and beams of light to measure the object but they are very costly and not always easy to move around with heavy scanner. To come up with good quality scan using triangulation method you just have to scan same area multiple time and areas with higher point density result in a higher mesh and higher accuracy. Triangulation method is less sensitive to ambient light. But this method also gives lower resolution and higher noise. This method often used on type of scanner which are built for small objects and only short distance away from the scanner. But the type of stuff you can scan with wary. However, the some stuff where this method will not work is if object has shiny surface. During the scan the object being scanned should be steady; Moving object can't be scan using this method.

One way to improve the method is improving the hardware. With minor improvement general triangulation can gather data with micron- level accuracy. Also improving type of color use for laser also affects the accuracy provided right sensor. Standard available triangulation sensors operate using read laser light at wavelength 670nm and blue laser operate on 405nm. Blue light is good if you want to scan like organic materials such as wood, skin, foodstuffs, etc. There has to also some requirements on the laser source that we can use, it has to be high beam quality in order to illuminate a line or small spot over a large distance. A certain optical power level is also required, particular for targets with diffuse reflection. For safety an eye safe laser wavelength in the 1.5um region. Red laser diodes with a power of a few mill watts are mostly use in triangulation. This method it doesn't have to laser line it can be done with laser dot as well but to improve the quality and time of scan. It is good to use laser line instead of dot. The cheapest way to create is laser line is to create your own. Laser dot can transformed to stripe of light when it pass through glass rod, it movable and can be adjusted horizontally or vertically.

3.7.2 Phase Shifting

Pattern projection techniques use multiple stripes or patterns projected simultaneously on the object, rather than scanning a single laser line or point on the scene. The popular method is the application of "Moire principle". Where a sinusoid pattern (or grating) is project on object with projector, and a same pattern again is projected with phase shift on to previously projecting pattern. It will create new third pattern shape with superposition. We don't necessarily have to have special pattern to be projected, there are methods based on a random pattern projection. But it good to have some structured light such as single stripe, multiple stripes, gradients, grids, gray code or bars. But sinusoidal works well. Then new pattern is captured with a camera from different angle. From a sequence of new patterns camera pixel the phase detected within one period of original pattern. In order to determine the absolute position within the superposition pattern the period has to be localized. Once the correspondence between the image field and the projection field is determined, the 3D coordinated information of the object is calculated with triangulation. Since its still using triangulation we still need to know positions of the camera and the projector. Unlike regular triangulation all the calculation can happen on parallel, this method is fast and fitted for semi real time scanning. To make it real time scanning we can improve same technique by use of a projection pattern and the detecting same pattern from multiple angles using stereoscopic system instead of just camera and projector system. This fast 3D scanning system combines the stereo and phase-shifting. Most of the high quality or high speed phase-shifting methods require and efficient phase unwrapping method (algorithm to calculate). Since it doesn't need to observe the undistorted measurements from direct surface reflections this method is little more robust against noise then other methods.

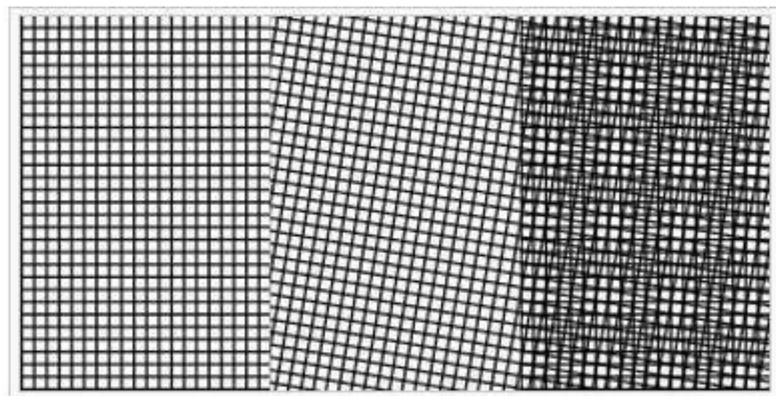


Figure 3.7.2 (1) Moire grid pattern
(Giving credit to Mathsisfun.com)

The accuracy of the measurement mostly depends on correct matching of the stripe orders of the reflected superposition patterns and on their proper connection to the corresponding orders in the projected patterns (calibration). For structured light method calibration need to be perfect; even with one percent error in calibration turn whole 3D model being out of place. The advantage of these scanners is the large resolution and accuracy obtained leading to high quality 3D surface reconstruction. The main advantage that batters then other methods is that the projected visual features are easily captured by the camera. We don't need large number of sample like laser triangulation we only need few sample, we can calculate lot of points with very few scanner and that way make this methods faster. Depending on how complex the pattern, it decide how many sample we going to need. Simple pattern like gray code, more number of sample going to need to acquire good quality mesh. We can also use color gradient which will require very few sample but we need good camera and but calculation get more tidies and more difficult interpret the captured images. But it's possible to capture color of the object using this method. Some down side of this method is length and height is fixed for this kind of scanner and costly due to need of projector and not portable. And it's more sensitive to surfaces and requires specific lighting. For scanning bigger object using this method it wastes of a lot of optical energy. For wide range object using laser scanner is best. One big problem is in order to scan the object in all dimension both the projector and camera must be rotted, or if the both the light and the camera are fix then or the object must be moved at each iteration to get the full 3D scan. Projector and object both being heavy it quality challenge to use this method.

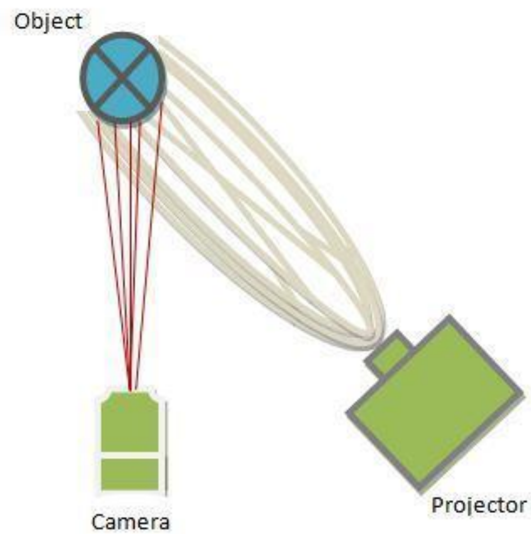


Figure 3.7.2 (2) Working principle of pattern projection technique (Credit to speigitalilibrry.org)

This method good for short range 1 to 5 meters, high precision full- field 3D imaging. It tend to work good for with small volume, full surface, need large point count, reasonably controlled conditions. For the projector light has to intense to project the pattern. Unlike laser projector use light bulb which is not very intense compare to surrounding light. This make nearly impossible to use scanner outside.

3.7.3 Stereovision

Stereo vision is a technique that uses two cameras to measure distances from the cameras, similar to human depth perception with human eyes. This method is very useful in portable camera. Stereo vision methods are known to be able to capture fine scale surface details with good hardware. The process uses two parallel cameras aligned at a known distance of separation. Each camera captures an image and these images are analyzed for common features. Triangulation is used with the relative position of these matched pixels in the images. Triangulation requires knowing focal length 'f' of the camera. The distance between the camera bases 'b', and the center of the images on image plane 'c'. 'd' is the difference between the side distances to the feature pixel 'v' on the image plane from center. Using the concept of similar to triangulation, the distance from cameras ‘D’ is calculated by focal length multiply by distance of camera base and dividing by ‘d’. Matching between pixels are happen by searching through the image and using methods like correlation or sum of square difference measures to compare neighboring pixel. Output of stereo vision system is an image of inverse range between images at each pixel. It is also possible to use multiple cameras like Artec 3d scanner, and compute image to image corresponded between image pairs and for each camera viewpoint.

Computer vision tries to copy human stereopsis by using two cameras as eyes in what is called passive stereovision. To make it more cheaper way of doing this is in using a single camera and moving it to different fixed positions for perceiving the scene from multiple points of view. This method is one of the important topics in computer vision because it allows the three dimensional position of an object points on the image planes. Instead of camera we can use electromagnetic sensor, usually work in same range as normal cameras. Where projected signal gets reflected from object and capture and with lot of calculation 3D point is created. Stereo vision is popular methods but the accuracy of the method depends on the feature detection and good software that can match botch images. And the smaller the baseline the more chance of making error in calculation. In 3D movie same technique use because its gives illusion of depth in 2D image. But this is different than 3D displays that display an image in three full dimensions. Applications for this method are lot such volume measurement, quality control, access control, tracking people movement, measuring shape of body parts, head movements. Unlike structure light this scanner can be use in outdoor environment. This methods is very fast and provide full 3D point cloud, its passive sensing so it doesn’t required much power. Some downside to methods is its very expensive in term of computational power. And usually requires solid stereo rig mounting and calibration. If we use camera instead of electromagnetic sensor there is sometime problem with shadows.

The big problem happens when it doesn’t know which parts of one image correspond to which parts of another image. Reason may be in due to movement of the camera, the elapse of time, or movement of objects. Determining which pair of projective points represents the same three dimensional object points. This is known as the correspondence problem. This can be limitation to stereovision method. There is method to solve problem but it doesn’t always work because it depend and what kind of object you scanning. The correspondence problem cannot be solved when observing non-textured objects, and points only appear in one of the images due surfaces. But a method is good for reconstructing dense 3D surfaces. Also it can merge with structure light methods and we can give unique code to every unitary position on the image by projecting structured light pattern onto the scene. This will give illusion of texture onto an object, and it will increase the number of correspondences. Then surface reconstruction is possible when looking for differences between projected and recoded patterns. Whereas above structured light techniques based on projecting simple primitives like pattern. Even without the problem using structured light make way easy to solve. It has one-shot image acquisition which makes low cycle times. It can provide higher resolution.

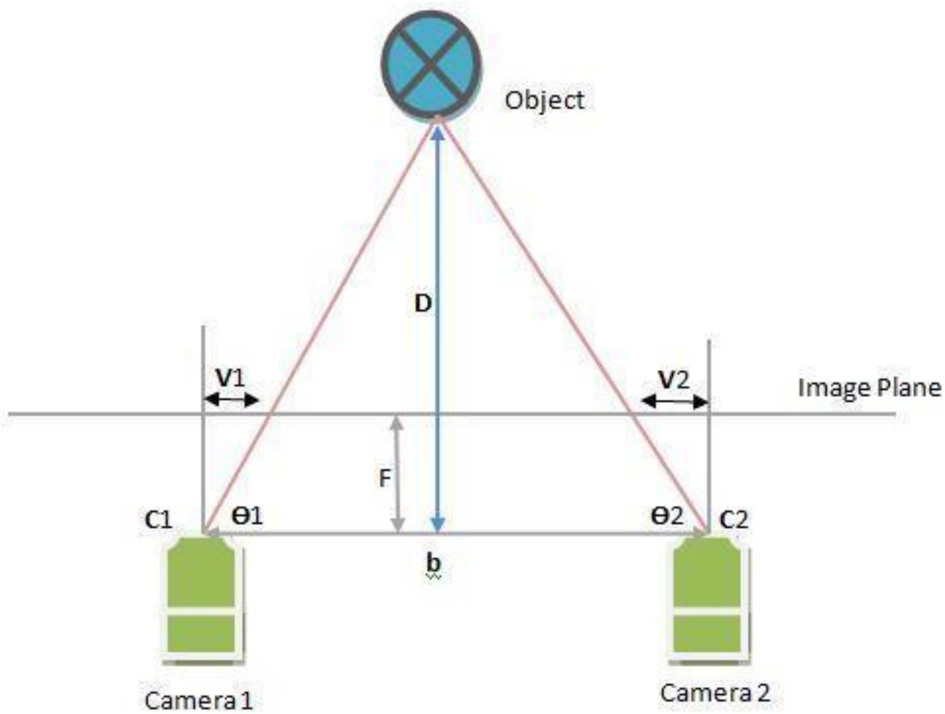


Figure 3.7.3(1) Stereo vision triangulation
(Credit to ni.com)

3.7.4 Shape from focus

The shape from focus method moves the object with respect to the lens surface and obtains a sequence of images that correspond to different levels of object focus. 'The sum-modified-Laplacian' focus operator is used to measure the relative degree of focus between images. The operator is applied to the images sequence to obtain a set of focus measures at each image point. The focus measure values at each point are modeled and interpolated to obtain accurate depth estimate.

The basic idea of image focus is that objects at different distances from a lens are focused at different distance. Shape from focus is the problem of reconstructing the depth of the scene dynamically until the focus of the camera is at point of interest. The change in the focus is obtained by changing either the lens position or the position of object, compare to camera. The point in focus gives information about its depth through the thin lens. Shape from focus is search method which searches the lens focal length that corresponds to focusing the object. In shape from focus, an object is moved with respect to the camera and a sequence of images that correspond to different levels of object focus is collected. A focus measure is computed in the small image areas of each of the image from in the image sequence. The value of the focus measure increases as the image sharpness or contrast increases and it attains the maximum for the sharpest focus image. So the sharpest focus image regions is detected and taken out. This simplifies auto focusing of small image regions by adjusting the camera parameters so that the focus measure gets the maximum value for that image region. Focus measure determine the focus value or sharpness of image pixel. Usually high pass filtering is used to determine sharpness. Also focused image regions can be synthesized to obtain a large image where all image regions are in focus.

This approximation method is applied to obtain a precise depth map. This method is slow because the focus value is computed for each pixel. To improve on it Gaussian model is fitted, it's a rough estimate is taken by applying the same method and then the best- focused values lying on image surface are searched. But it needs more computational time. First an initial estimate obtained by applying a focus measure locally, and then depth map is refined through an approximation method. The performance of these methods largely depends upon the initial estimate. Since there is one to one correspondence between object point and focused point in image, so this method is very sensitive to noise. Also to have three dimensional points of whole object, we have to take a sequence of images at different focus levels. If there is problem with focusing between two connective frames final result may not be accurate or any other kind of problem that can cause to acquire its Focused image surface.

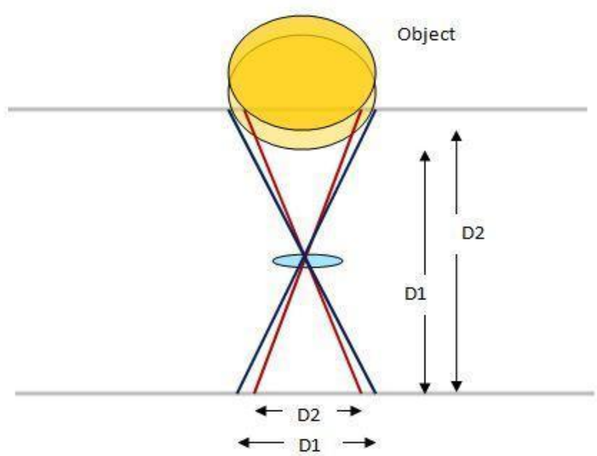


Figure 3.7.4(1) shape from focus
(Permission pending)

In figure at bottom all the rays emitted by the point P of an object and intercepted by the lens are refracted by this one and converge at point Q in the image plane. The position of the point P and its focused image Q are related by the Gaussian lens formula. Where the f is focal length, u is the distance of the object from the lens, and v is the distance of the focused image from the lens. In the figure a focused image is sensed by the image detector (ID) if the image detector matches with the image plane. If distance between the lens and object equal to the distance between lens plans and converge point; then a sharp image Q of the point P is formed. If a point light source placed at point P is not in focus, then it gives a blurred image on the ID plane.

The degree of blur depends on the focal length f of the lens, and the distance u and s. If focal length and v is known, then the distance u of the object can be founded. In conclusion, the depth of field decreases when the focal length increase, in the same way, it increases when the diameter of the circle of confusion increases which depend on type of camera being use. This depth of field is directly correlated with the depth resolution of the 3D reconstruction.

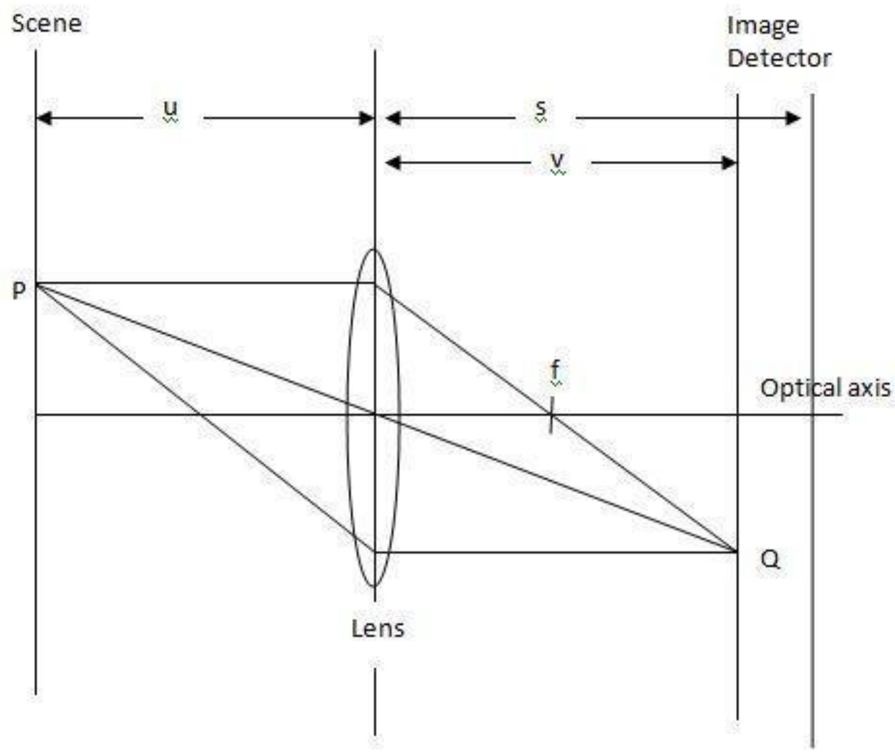


Figure 3.7.4 (2) Shape from focus (Convex Lens)
(Credit to sipl.gist.ac.kr)

3.8 Data Representation Software

In the past few years' data representation software have been of great help to visualize data obtained analytically and systematically. Specifically, in the 3D world it has helped the people to obtain smooth and rendered model of object captured by lasers and cameras. Software visualization especially in the 3D Scanning world has revolutionized in the past few years. Different tools and techniques have been developed to address many problems. However, the road is not really clear, and there is still a lot of work to do in this matter. These techniques offer a wide range of solutions for particular problems. One of the most successful and well-known application for data representation is SeeSoft. However, back in the 90s when this application was created, little or none was known about dynamic data representation software, especially in the 3D Scanning world. Back in the day, the idea of scanning an object was merely reduced to 2D, this means that scanning was reduced to represent data in the x and y coordinates.

After many years of research and learning different methods and techniques to represent raw data independently on how it was obtained, engineers and computer scientists developed different approaches to actually implement complex mathematical algorithms in code to model 3D objects based on distance, shape, size, dimensions, etc. Most of these techniques came to place to make the 3D scanning world more enjoyable and actually less frustrating to achieve. Our focus in this project is to research the visualization of small scale software to assist and analyze small objects placed more or less one meter far from laser and camera tools. This thesis brings together research from different 3D data representation software as well as human and computer interaction.

We would like to use different data representation software to see which one is the best for our needs. We will try to convince the end user that our product is the cheapest to make in the market and can be easily done by just buying the parts and assemble them. Our source code will be available online for free or it will be open source which will allow the most experienced users to actually edit and change the source code of the data representation algorithms to their needs. There are many goals to actually succeed in this project, and that is why we will focus on the different software visualization systems, the kind of users, and the available tools in the current market. Two criteria will determined the success of this project. One is the expressiveness which refers to actually represent the modeled object that we want to visualize in the data representation software. The other is the effectiveness of representing the object. This latter refers to actually debug the code to a less robust and less complicated algorithm, as well as reducing future maintenance of source code.

Finally, considering this and more criteria, we would like to learn not only to represent object in three dimensions but also, we would like in the future, to enter the world of 3D printing, which is kind of vague and not so reliable. However, for now we will focus on getting raw data images from laser and cameras, and translating those files into viewable smooth rendered 3D models in the best software available out there, and the ones that we feel more comfortable with.
We will discuss different types of 3D data representation software in the following pages. Some of them are pretty robust programs, and coding intensive. Some are really easy to use and can be applied to different applications. However, we will focus on 3D application.

3.8.1 Open CV

OpenCV is a data representation software mainly focused on computer vision. The functions in this section use a pinhole camera/laser model. In this model, a scene view is formed by projecting 3D points into the image plane using a distinct algorithmic techniques.Real lenses usually have some distortion, mostly radial distortion and slight tangential distortion. OpenCV extract 3D models of objects, produce 3D point clouds from stereoscopic cameras, stitch images together to produce a high resolution image of an entire scene among other features. OpenCV is mostly written in C++. However, we will try to code the algorithm to control and get images in Java.

Here is an example of OpenCV 3D model as well as its respective rendered object. Also, notice that to actually scan/laser the object, a series of algorithm must be in place such as calibrate the camera, size of object, dimension, shape, etc. Calibrating the laser/camera in order to get the best view of the object that will be modeled in 3D is the most important aspect when using OpenCV data representation software.

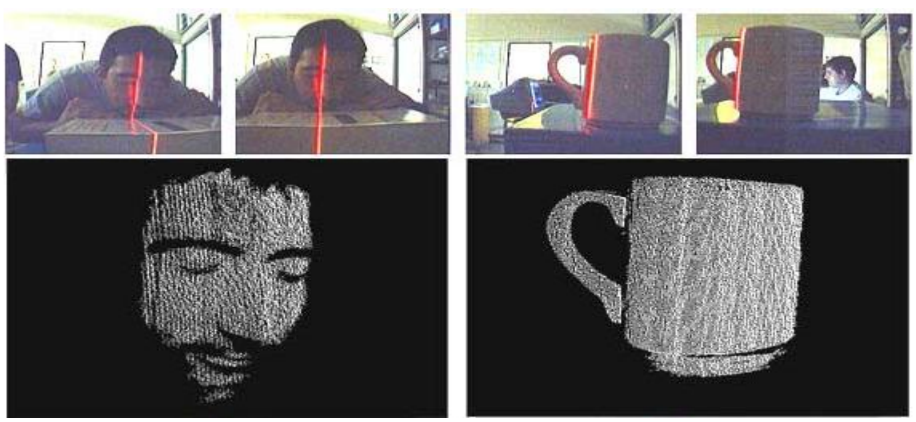
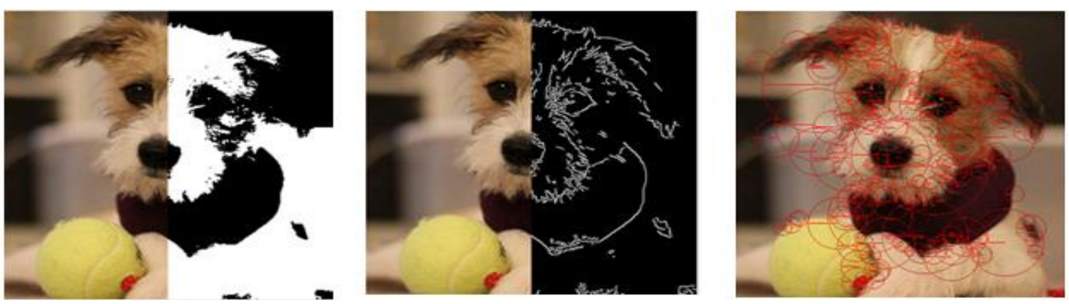


Figure 3.8.1 (1) Open CV Sample
(Credit to stackoverflow.com user)

In this image we can see the laser and camera scanning two objects by active scanning which means that object is standing still while laser and camera scans. In contrast with passive scanning which means that object will scan in a single spot without moving nor pinpointing each and every spot of the object. This is not desirable, we want to properly get every single point in the object in order to have more data to then model it more accurately and effectively. Finally, OpenCV is just one of many different data representation software out there. It is a really good software to scan and model 3D objects. However, its applications differ widely.

3.8.2 Simple CV

This data representation software is just a less powerful OpenCV mainly focus on specific applications on computer vision. It is mainly written in Python and its applications are easier to understand while keeping a good overall 3D object result. As they say, Simple CV is just computer vision made easily. Here is an example 3D scan using Simple CV written in Python as well as different techniques such threshold, edges, and key points method.



A **B** **C**

This image shows the key points method which pinpoints most important aspects of the original picture and makes it noticeable in the final object model to further analysis.	This image shows the set edges method which converts and pinpoints every pixel of the original picture into a white rendered model with a black background.	This image shows the threshold method/function of SimpleCV and how it translates/models the original picture to a black/white gray scale of the rendered model.

Figure 3.8.2 (1) Simple CV Sample
(Credit to simplecv.org)

3.8.3 Autodesk 123D

This 3D data representation software has been used widely because of its simplicity and effective way of modeling 3D objects taken from various sources. Within its applications are modeling objects, catch objects, design, create, draw objects, etc. It is also free to use with a paid premium membership which allow its users to access more and more features which might increase the quality of a determined task that they need. This and more 3D modeling software are available in the market, but Autodesk is highly dedicated to the world of computer vision.

3.8.4 SolidWorks

This software is a computer aided-design meaning that most of the objects modeled are created by end user. SolidWorks since its creation in 1995 has been widely used in the computer vision industry as well as many other engineering fields to recreate specific objects based on parametric characteristics. We will not be using this software; however, it is highly used in the 3D world to visualize better objects. Note that this software is not free to use and it costs a lot of money. However, over 2 million engineers use it daily for their tasks.

This and many more software have been widely used for the past few years to model and recreate specific 3D objects. This does not mean that the road has ended. The world of 3D modeling is still young and a lot of work is still required to perfect the end result of a successful modeled and rendered object.

3.8.5 3D Scanning Approaches

There are a lot of scanning approaches in the 3D modeling world. Mostly, they are based in the range of distance that the 3D scanners will be placed. They are based on different principles of imaging. Some scanners are ideal for short range scanning, others for medium range and long range scanning. This mostly depends on the capacity of the laser/camera. For our thesis we will be focusing on short range scanning meaning more or less 1 meter far from the scanning device. We will discuss furthermore different approaches on 3D scanning as well as their pros and cons.

3.8.6 Point Cloud

This is probably the most important 3D data representation out there and the one we will be using and playing around with throughout this whole thesis. Point clouds are really important to represent modeled object since they give a rough sketch of how the objects will look like digitally as well as represent the characteristics of the scanned 3D model. It is a set of data points in the coordinate system. Specifically in the three dimensional coordinate system, these points are normally defined as {X,Y,Z} coordinates, and they are used to represent the external characteristics of the modeled object. These point cloud are created by a regular 3D scanner. The 3D scanners are really useful in creating these points to model 3D objects. Normally, the scanners get the input file or point clouds and translate these to a data representation software which gets the output file and models the object.

Here are some examples of how point cloud look like after they have been scanned and rendered in the data representation software. We will be dealing with many point cloud in this thesis in order to get a better understanding on how these are really helpful in representing and visualizing the desired scanned 3D object.

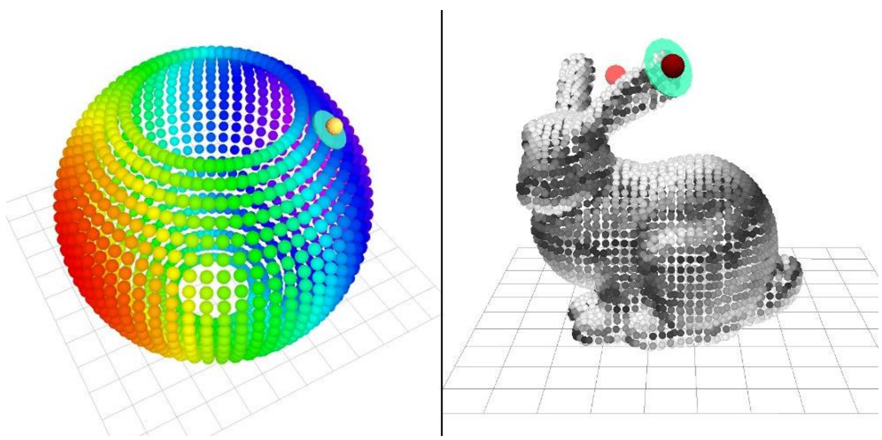


Figure 3.8.6 (1) Point Clouds
(Credit to Stanford University)

There are many uses for this point clouds; however, its main use is to model 3D objects as well as giving the accurate shape, color, dimensions of the scanned object. There are many software that deal in representing this cloud points. One of them is RenderManStudio which automatically handles most of the setup for backing up point clouds automatically. RenderMan Studio automatically takes care of all the background work for outputting a point cloud for us. It generates a rendered scene and outputs the point cloud data as a .ptc that contains the information and characteristics of the modeled object.

Finally, this point cloud can be in color, black, and white gray scale. It is the best approximation of the original object modeled digitally. We will be limited to small size object in this thesis for the purpose of practicality, and the fact that we will be using a small range laser/camera.

3.8.7 Short Range Scanning

This type of scanning approach is based on laser triangulation and it will be the approach that we will use for our 3D scanner. We will dedicate a whole section explaining the details of this kind of scanning. For now, let's explain how it works merely. Laser triangulation scanners calculate the distance from the scanner to the object using a laser line or single laser point across the object. It is compose mainly from a laser and a camera to capture the image and a sensor to detect that an object is going to be scanned. The distance between the object as well as the angle to which it will scan it is known very precisely. It also detects the shape and size of an object following some algorithms that we will explain later on. There is also another type of short range scanning approach which is the structured light approach. The only difference between these two is that the laser triangulation approach actually scans the object with a single line while the structured light approach scans the object with several angles and multiple laser lights. Short range scanners are really useful to model small object and recreate them in a short period of time digitally. This is our main approach and the one we will be using throughout this whole thesis.

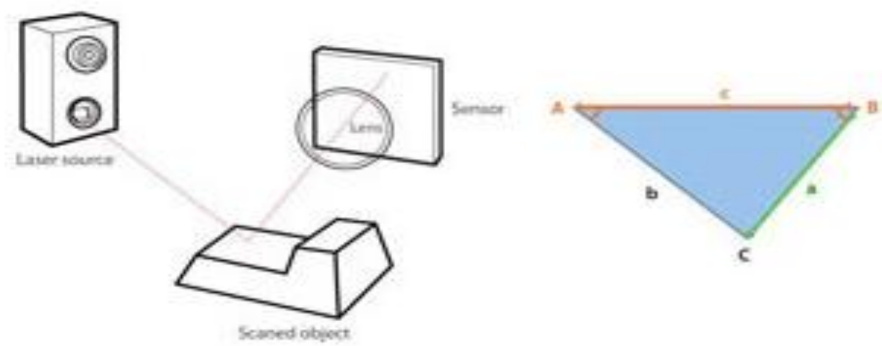


Fig 3.8.7 (1) Laser Triangulation Approach
(Credit to rapidform.com)

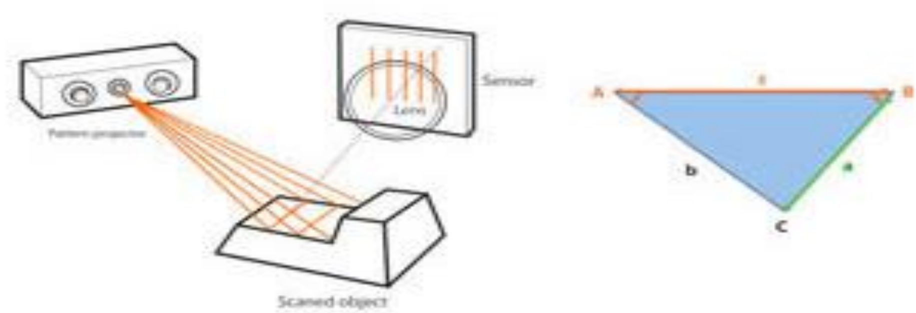


Fig 3.8.7 (2) Structured Light Approach
(Credit to rapidform.com)

Cons	Pros	Short Range 3D Scanners
Normally not so accurate, generally lower resolution, Higher noise when scanning.	Available in many forms, portable, easy to assemble, handheld, less sensitive to ambient light.	**Laser Triangulation**
Limited to scanner area, not so portable, very large, very sensitive to ambient light, and might require a specific surface.	Very accurate, higher resolution, lower noise.	**Structured Light**

Here are some detailed images on how these short range scanners look like and how they work while scanning objects and representing them in a data software 3D modeling.

3.8.8 Medium-Long Range Scanning

This is another type of scanning approach which is really useful to scan larger objects and farther away, normally from 2 meters to a 1000 meters. As a matter of fact, this kind of approach requires a more powerful laser and camera to scan the desired object. Several projects have been done with this type of scanning approach.

However, we are trying to minimize the cost of our 3D scanner; therefore, we will not be using this approach. Ultimately, this mid/long range scanners can be divided into two types, a pulse based approach and a phase-shift approach. The first is based solely on the speed of light meaning that the laser/camera takes in consideration the distance to which it is going to scan. An object is placed far away and the sensor/laser gets this reflection of light and outputs the desired file. The second phase-shift approach works similar as the first one. However, it also takes into consideration the frequency wave of the light reflected from the object.

These are really good 3D scanning approach for long distance, however, as we mentioned they are relatively expensive and kind of large, not portable, as opposed to short range scanning devices. Here are some examples images on how these scanners work as well as a table for their pros and cons worked similarly in the short range scanning section.

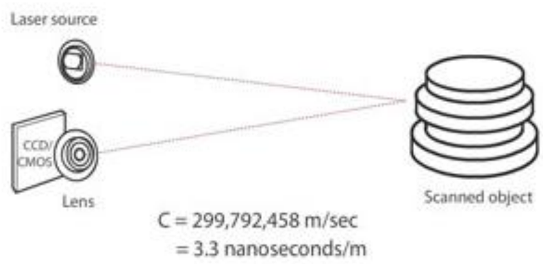


Fig 3.8.8 (1) Pulse based approach
(Pending permission from rapidform.com)

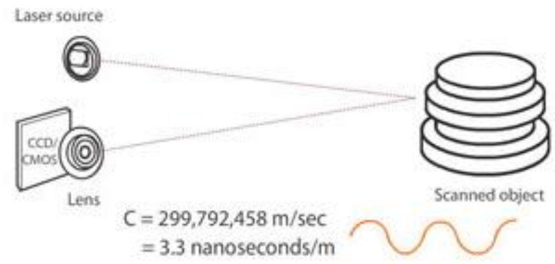


Fig 3.8.8 (2) Phase-shift based approach
(Pending permission from rapidform.com)

Cons	Pros	Mid/Long Range 3D Scanners
Not so accurate. Not so reliable. Takes more time to obtain output file. Kind of noisy.	2 to 1000 meter scanning far from object.	**Pulse based Mid/Long Range**
2 to 500 meters scanning far from object.	Very accurate, data is obtained faster, low in noise.	**Phase-shift Mid/Long Range**

4.0 3D Scanning Implementation

When we talk about actually implementing the 3D modeled object in the data representation software we are planning to get the input file from the controller unit and then pass it through the software and algorithms that we will be implementing. Some major issues that we might encounter along the way is how to completely model and render the scanned object. We are planning to write some algorithms for each part of the thesis. Now, we want to implement some major mathematical and modeling algorithm starting from controlling the servo motor, rotating the camera and laser, and finally, the most important part that is actually modeling and rendering the scanned object. The controller part is the second to last most important part of this project. We will dedicate a whole section talking about this issue. After obtaining the final input file from the controller via USB/FireWire cable, we will begin to pass the input file to the algorithm to finally display it in the data representation software that we have been discussing above. This and more important aspects of our project will be put into place later on when we start building and coding our project. For now, we have a pretty solid background programming wise on how this 3D scanner is going to work computationally and electronically.

4.1 Functionality

The main functionality of a 3D scanner is to create a point cloud on the surface of the subject. The point clouds are later reconstructed by extrapolating the points in the surface of the scanned object. Color can be also obtained if the correct algorithm and camera/laser synchronization is in place. We are planning to recreate/model the object in gray scale. However, if we find that modeling the object in color is within our potential, we will gladly model and render the object in full color.

Some 3D Scanners are really related to high definition cameras since they are used to capture the image and send them to the pc for further use. The similarities between cameras and 3D scanners have to do with that they both have a cone like field of view and they can collect information of surfaces of objects that are not obscured and have a greater field of light to make reference to so that they are not opaque. Cameras collect color information about surfaces within the field of view. Also, they collect distance, and other aspects of the object to be captured like dimension, shape, size, etc. The pictures obtained by the camera are therefore modeled like points or point clouds and this allows for 3D dimensional analysis as well as modeling and rendering the desired object.

In most cases, a single scan won’t produce the complete desired object modeled since there can be many factors that might not contribute to its correct capture. Many scans of the same object, even thousands of hundreds might be require obtaining every single desired point cloud to correctly model the object. Also, it is necessary for the 3D scanners to scan every single side of the object. This can be obtained by a process called alignment of the laser along with the camera so that it checks for every single possible spot of the object. Then, a complete modeled object is obtained for further use. This also contributes to the creation of a model as whole rather than just single unspecified points of the objects’ surface.

There are many more functionalities of 3D Scanners which we will discuss further in another time. For now, let’s focus on the basic one which is just to get the object modeled by a simple laser and camera set up to then renderize it in our data representation software. We might be able to obtain several thousands of point clouds of a single object and model them correctly. Finally, there is a still a lot to do in this matter, we are talking about many years of experimenting with it.

4.2 Technology

Multiple technologies can be implemented in 3D Scanners, however, many of them have limitations, advantages as well as different approaches, and we will discuss some of them along this thesis. Many technologies like optics and spectroscopy can implement 3D scanner functionalities since it is all about getting that information captured by the laser and the camera. Collecting the data can be useful in many applications. For example, in entertainment world many object need to be model in animation movies.

Many industries like Pixar spend a lot of time and money to obtain data and render/model it for the movies. Video games as well used this technology of data scanning and animation. Consumers pay for good quality movies and video games; therefore, enterprises spend a lot of money annually to offer end user the best experience available in the market. Other technologies can be applied to industrial purposes, design, medicine, and other fields of engineering, control processing, visual aid processes, and culture.

4.3 Reconstruction

Point clouds produced by 3D scanners are used to measure and visualize in architectural and construction to better identify fallacies and properly design and model better solutions to problems that might arise in this industry. Many applications well discussed throughout this thesis will implement many aspects of the reconstruction of the desired object.

3D scanning devices use point clouds to represent objects. However, this point cloud are well modeled in the form of polygons, surfaces like NURBS (Non Uniform Relational B Spin line) which is a method used in computer graphics to correctly identify and model curves and surfaces, also there are many others which we will not discuss since they are not really relevant to this matter.

For now, we will try to explain some characteristics of reconstructing the data obtained by the laser and camera as well as the controller and software programming unit. Finally, 3D reconstruction is the process of capturing all the characteristics of an object like its shape, dimensions, colors, etc. to recreate this model in real digital life. In the following pages we will discuss some types of 3D modeling reconstruction and how they are use in real world applications as well as making reference to previous works done in this matter, and how they programmed their devices to scan different objects.

We will try to use as many possible scans in the minimum amount of time to properly reconstruct small object which is the intent of our scanning device (short-range). In the following page we will represent some images of reconstructed scanned objects and its respective synchronization laser/camera/data representation software.

Fig 4.3 (1) 3D Car Reconstruction
(Pending permission from MIT Lab)

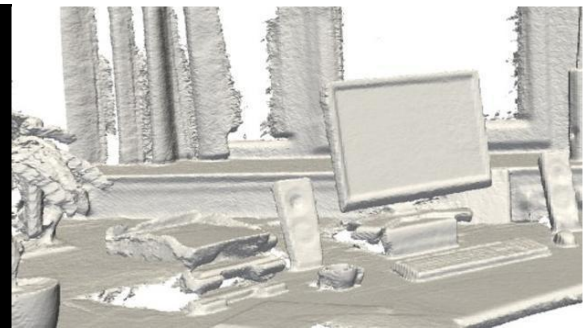


Fig 4.3 (2) 3D Computer Desktop Reconstruction
(Pending permission from MIT Lab)

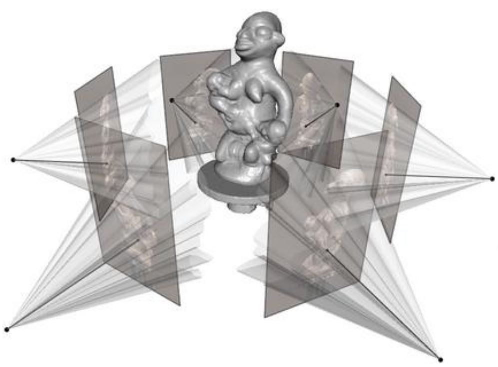


Figure 4.3(3) 3D Ancient Statue Reconstruction
(Pending Permission from vision.in.tum.de)

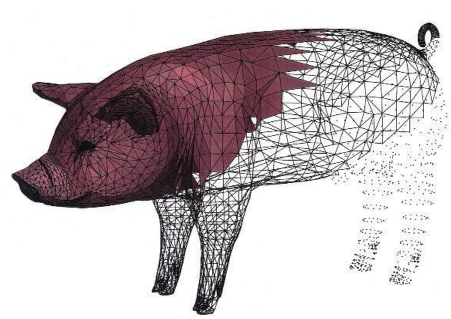


Figure 4.3(4) 3D Pig Reconstruction
(Pending Permission from vision.in.tum.de)

In the following pages we will discuss some major methods of reconstructing 3D objects as well as their different output or image processing representation. We will begin by giving a small introduction from a mathematical point of view and how these object's shape fit perfectly in a modeling mathematical scheme. Some are polygon meshes, surface and curves models and other CAD models.

Polygon Mesh Model.

When representing a polygon shape as well as a curved surface many characteristics of an object took place. They are modeled as contiguous flat surfaces within the objects surface. Polygon models which are also called Mesh models are used for visualization of a specific objects' characteristic. 3D Reconstruction to polygonal models involves obtaining large data sets of points and interconnecting these points with continuous lines in order to create a complete model surface of the object. Many software are used to reconstruct these polygons, some of them are MeshLab, PointCab, etc. In the following pictures we can see some of these polygons modeled in the objects surface to better approximate the objects size as well as their external characteristics.

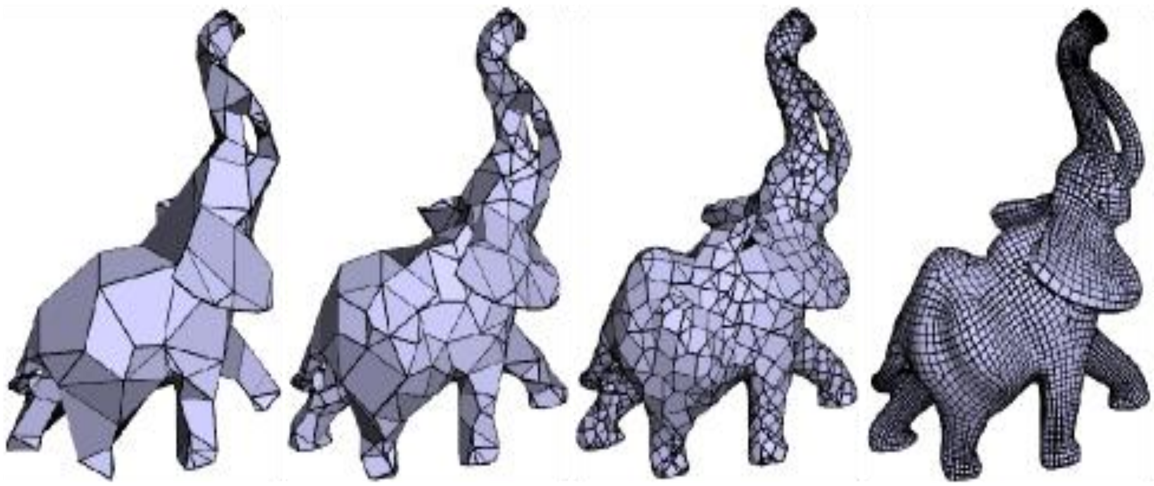
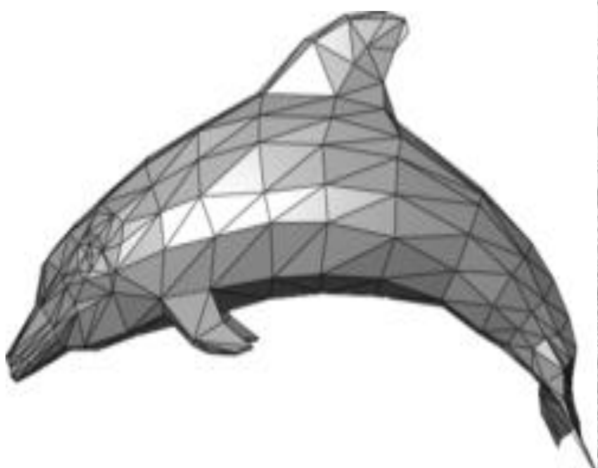


Fig 4.3(5) 3D Polygon Model of an Elephant
(Credit to MagSoft)



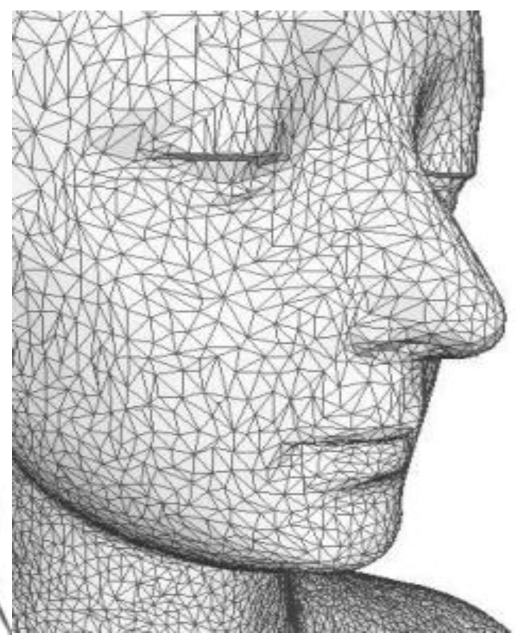


Fig 4.3(6) 3D Polygon Model of a Dolphin **Fig 4.3(7) 3D Polygon Facial Reconstruction**
(Credit to DeviantArt) **(Credit to ThunderCloud Studio)**

Surface Model - When we talk about surfaces model we are really referring to implement the scanning in huge surfaces like a volcano or a crater. Also, digital modeled curves are the best way to animate these surfaces. We will see some of them in the following pictures and how they assemble some of the most precise modeling techniques for huge dimension objects. Also, notice that the most important level of sophisticated modeling techniques involves the use of quilt curved surfaces which patches to model a desired object.

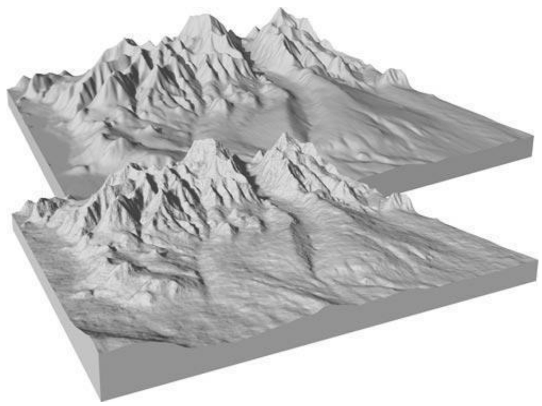


Fig 4.3(8)3D Surface Model of a Valley
(Credit to pubs.usgs.gov)

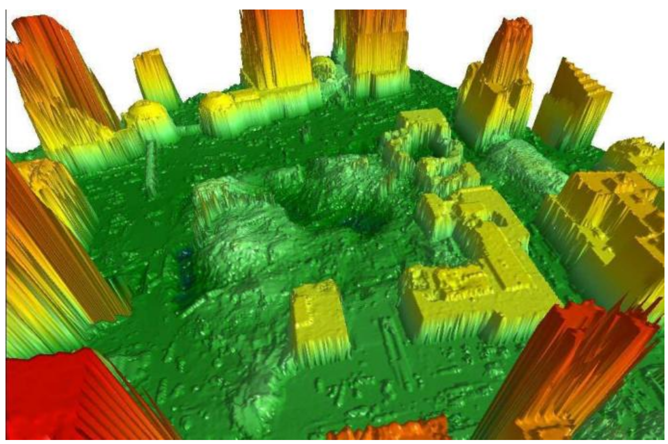


Fig 4.3(9) 3D Surface Model of Aerial New York City
(Credit to GIS Community)

Solid CAD Model - Solid CAD modeling consists basically on a set of mathematical and computer aided visual modeling scheme to represent 3D solid models. This differs basically to some basic knowledge in geometrical and computer aided design by referring to its physical characteristics rather than the actual complete modeling scheme. This altogether with computer graphics and virtual computer aided design support the creation of well relative 3D models such as visualization, animation, and digitalization of modeled physical objects.

Solid modeling is really useful when dealing with mechanical parts since they help designer and end user to better visualized objects in three dimensions. Also it is important to note that this type of modeling method is really good in speed design and detailing for manufacture since they give a rough idea on how this model improves visualization as well as communication, and it eliminates also design interference issues.

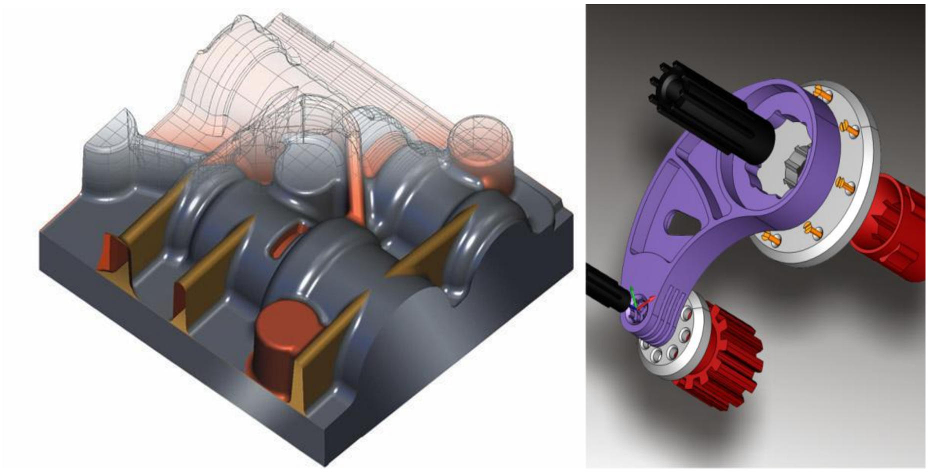


Fig 4.3 (10) Solid CAD Models
(Credit to cadcamsoftware.com)

4.4 Power Supply

4.4.1 Voltage Regulation

After the rectification stage, the AC signal will need to be regulated to generate a DC signal. Concept of regulation. Voltage regulation will be needed for other parts of this design including the microcontrollers, platforms and the DC motors. Most electrical power systems are prone to slight variations. Almost every electronic circuit has been designed to operate off some given supply voltage, which most of the time are assumed to be constant. A well designed voltage regulator would provide the required constant DC output voltage needed. The entire design would contain circuitry that would continuously hold the output voltage at the design value without worrying about any issues relating to changes in load current or input voltage keeping in mind that the load current and input voltage are within the specified operating range. With maximum efficiency as one of our aim in this entire project, choosing the best voltage conversions will be a priority.

Converting the main supply of 12-24V DC supply input to the system down to the required voltages for the other parts can be done by considering using the abundant regulators that are available on the market but the most used ones are the linear based voltage regulator and switching based regulator. The most basic building block of almost every power supply used in general electronics would be the linear regulators. Considering conditions such as Maximum Load

Current, input/output voltage ranges, tolerance, idling current and other factors will be noted

One of the easiest way's in reducing a DC signal is by either using the linear regulator in an integrated circuit (IC) form .Voltage regulators actually do not produce any power, they rather consume some of the input power based on their rated efficiency rating. Commonly known types are the T0220 package which consist of a three terminal IC. These IC's have protruding legs from a plastic case with a built black metal plate around it, in order to bolt it to a heat sink. The LM78xx which are usually positive voltage's and LM79xx which are negative voltages are commonly fixed voltage solid-state regulators which has the last two digits of the device number indicating the voltage output. This helps in easier identification to the regulator. We are considering using the LM78 series.

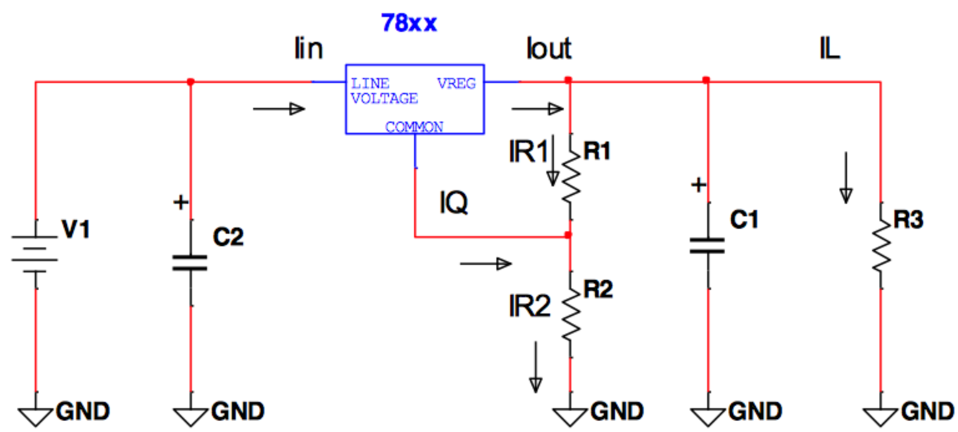


Fig 4.4 (1) LM78 Circuit Diagram

Fixed output IC regulators could be designed to be more adjustable. Two of the commonly known ways of doing this are below:

1. Either a zener diode or resistor could be added between the ground terminal of the IC and ground being used. However if current from the ground is not constant, the resistor will be avoided. Switching in couple of different values for the components, the output voltage could be made more adjustable in a clearer step-wise fashion.

2. This approach will be function by placing a potentiometer in series with the IC's ground, the output voltage can be varied after this.

4.4.2 Zener Diodes

Zener diodes are other types of linear regulators. These are basically the standard PN junction diode but are specially designed to have a low pre-determined Reverse Breakdown Voltage that takes advantage of this high reverse voltage. The *zener diode* is the simplest types of voltage regulator and the point at which a zener diode breaks down or conducts is called the "Zener Voltage.

For optimum operation of this circuit, couple of cautions will have to be taken into consideration, including the power being dissipated will not be made to exceed the maximum rated value. When the current flowing in the diode is at a minimum, the driven load current will be at a maximum, and the source voltage will be a minimum. The opposite or vice versa should be applicable as well. A parallel arrangement with the zener diode is made with the load and the voltage source together with the diode is arranged in series with the regulation resistor. To be able to generate a constant voltage over the zener diode, the current should be adequate to drive the zener diode into its breakdown region. The output voltage will be constant even with a varying output load resistance and the ripple input voltage from the rectified AC signal. The minimum current being derived from the built circuit should be more than the minimum zener diode current.

BZX55 Zener Diode Power Rating 500mW							
2.4V	2.7V	3.0V	3.3V	3.6V	3.9V	4.3V	4.7V
5.1V	5.6V	6.2V	6.8V	7.5V	8.2V	9.1V	10V
11V	12V	13V	15V	16V	18V	20V	22V
24V	27V	30V	33V	36V	39V	43V	47V
BZX85 Zener Diode Power Rating 1.3W							
3.3V	3.6V	3.9V	4.3V	4.7V	5.1V	5.6	6.2V
6.8V	7.5V	8.2V	9.1V	10V	11V	12V	13V
15V	16V	18V	20V	22V	24V	27V	30V
33V	36V	39V	43V	47V	51V	56V	62V

Fig 4.4.2 (1) Zener Diodes Comparison

$$R_i = I_i + (Vs - Vz)/I_z$$

$$P_{zmax} = V_{zmax} * I_{zmax}$$

Izmax /20 <Izmin

$$Izmax_i = I_i\,(Vsmax - Vz) - Iimin(Vsmin - Vz)/Vsmin - 0.9V_z - 0.1V_{smax}$$

Since engineering is all about improvements we considered how to make the zener regulator extra better in terms of performance and this can easily be achieved by simply adding series of voltage regulators. The values of the individual Zener diodes could be chosen to make the design, meanwhile the silicon diode will be dropping to about 0.6 to 0.7V as far as we use the forward bias condition. The supply voltage, must definitely be high.

From the circuit below, connection is made between the load current, transistor, and a zener diode. The base current will now form a load current for the zener diode. This current is lesser than the load current and the output voltage will be around 0.6V to 0.7V smaller than the zener due to the transmitter voltage drop. This will now make a load on the zener, making the variation effects very small. There is still load sensitivity. The circuit is designed to have a series arrangement due to the fact that the regulating elements are made to be in series connectivity with the load.

Ri helps in determining the zener current and this could be derived by the procedure below. The minimum acceptable DC current gain for the transistor will be hFE min and the value of K will be made equal from 1.2 to 2 to make sure the Ri is low for IB

$$R_i = (V_s - V_z)/I_z + K \cdot I_B$$

$$I_B = I_l / h_{FE\,min}$$

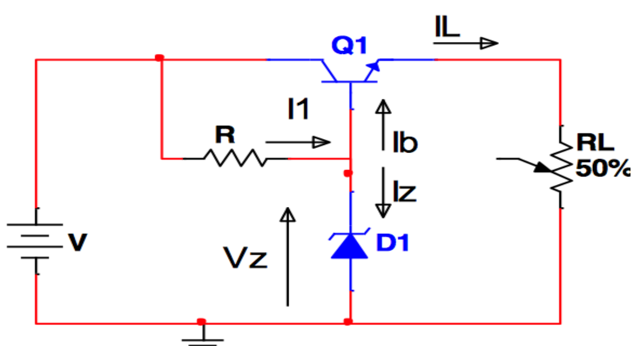


Fig 4.4.2 (2) Emitter Follower

In addition to generating a single stabilized output voltage, zener diodes could be made to connect with each other in series together with a normal silicon signal diode to generate a variety of different reference voltage output. All in all a zener diode will be operated in its reverse biased condition. The stabilized output voltage is always selected to be the same as the breakdown voltage VZ of the diode

Choosing between either a linear regulator or a switching regulator is a critical step since them both some advantages and disadvantages associated with their operations. Hence both regulators might be considered practically till final needs are met. A very common disadvantage from the system regulator is an unpleasant sound and series of waves from the system regulator, which is highly due to the switching rate.

Switching systems have so many advantages when compared to linear based regulators. In situations where there is a greater value of the input and output voltage difference, efficiency of the linear regulators and switching systems when compared shows the linear regulators being lower. An example would be when the input voltage is 12V and the output voltage being from 1 to 5V. An enormous amount of heat is generated on the regulator since a general rule of thumb requires the input current being the same as the output current.

4.4.3 Switching Systems

Although switching systems have a lot of advantages, a disadvantage exist which includes more external components which could be a burden as compared to the linear regulators. Switching systems may need diodes, inductors, and a filtering capacitor before the integrated circuits can receive an input. Linear regulators do not share this component.

They may need just a bypass capacitor. Widely considered for stepping down output voltage from the input,linear based regulators are commonly known for such purposes. However switching systems are widely known for a lot of purposes. They can operate in either stepping up, stepping down or even inverting. Such conditions don’t exist in the switching regulator.

4.4.4 LM Series Regulators

The LM series of linear regulators are very popular on the market in the application of smaller embedded projects due to their over current protection which are included in the design which then makes the drawing of excessive power much limited. This is partially due to the built in overheating protection which can regulate how much power can actually be drawn through the regulator at any given time.

With consideration of using a regulator for the use in stepping down the estimated power delivery of 12V down to 5V required for a portion of this project, several regulators exist, which could be considered. Texas Instruments has a considerable number of regulators but the LM's are being considered at this point.

This will include the 5V, 3.3V needed further explanations are being made in design section. The LM338K IC is a 12V regulator that has the capacity in producing a constant supply of 12V output and is configured to produce enough high output current in order to provide power to the rest of the components. This has the ability to provide a 13V/5A power supply The IC has time dependent current limiting, thermal regulation and is available in 3 lead transistor package. The IC can supply well over 5A at an output voltage range between 1.2V and 30V.

In this circuit the output voltage is determined by the two resistors R1 and R2.

The output voltages can be made to vary by adjusting the R2. Diodes D2 and D3 serves as protective diodes and finally the Capacitors C1 and C5 are filtering capacitors while C2 and C3 are known for decoupling. The LM7805 is a 5V voltage regulator that will take supply input of 12V from the output of the designed 12V part. Finally, the LM3X is a 3.3V voltage regulator that takes its input from the 5V section.

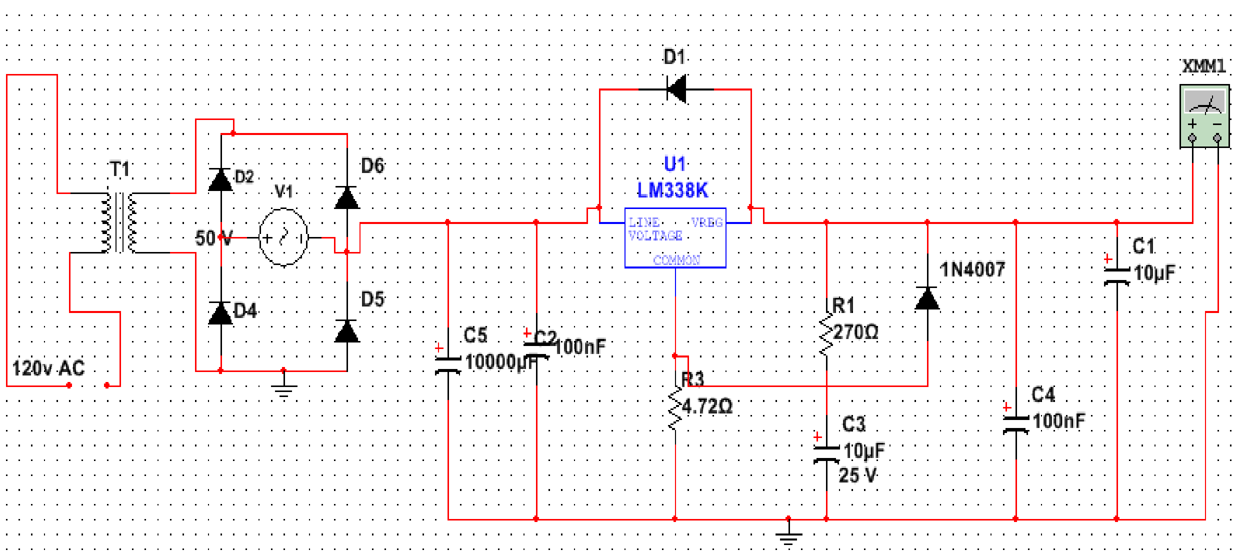


Fig 4.4.4 (1) LM Series Regulator Circuit

LM78xx is a product from the Texas instrument company and is being considered as a part of our power modulation. This regulator functions as a step down and is considered as a linear regulator.

The LM78xx has the ability of being able to provide an output total current of about 1.5A whilst the voltage is being kept around 5V.

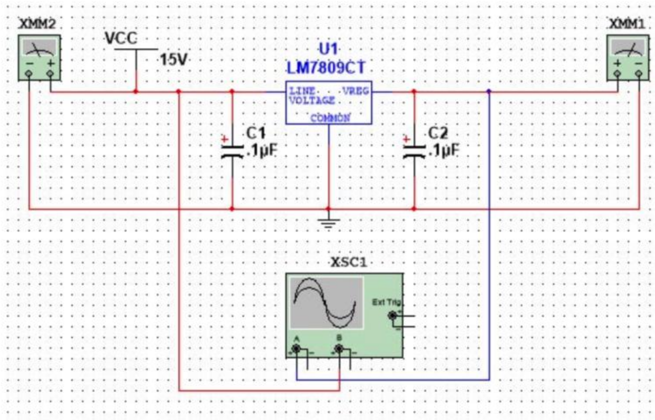


Fig 4.4.4 (2) LM78xx Circuit Diagram.

The 3 terminal voltage regulator from LM 7805 is shown below.

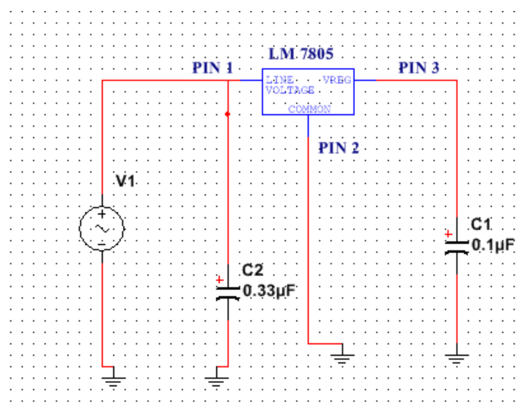


Fig 4.4.4 (3) LM7805 Circuit Diagram

Pin 1 functioning as an input Pin. For optimum regulated output voltage, in order to achieve the desired voltage. The LM7805 should be designed in such a way to receive a DC signal. We researched that the incoming voltage must be slighter higher than our desired voltage.

Pin 2 which functions in such a way for our circuit to have a return path and for the voltage to have an electric potential and as a ground and is fixed onto the circuit.

Finally, Pin 3 which verifies and shows our voltage we are looking for. This is referred as an output pin. It has been verified from Texas Instruments that the LM7805 comes in a plastic package with a total count of three pinned arrangement. The package has been designed to a heat sink for the dissipation of heat. This is not the only arrangement as there could be horizontal mounting package as well.

This is very advantageous as it can be mounted on the printed circuit board being designed for easy removal of heat generated. For a higher power requirements, a mounted heat sink is to be added to the board for the removal of the excess heat generated. This is done to provide safety for the components running from the rail of the regulator. Since one of our aims is to reduce the cost of this project, this linear regulator can be more economical due to its cheaper price as compared to the simple switcher outlined below.

4.4.5 TL 780

Information provided from each of the fixed-voltage regulator in the TL780 series has the capability of being able to provide 1.5 Ampere of desirable load current. A technique of providing a temperature-compensation, in addition with a trimmed band-gap reference which is designed internally, has been able to provide an efficient accurate measurements as compared to other regulators being discussed in this project.. Advanced layout techniques provide excellent line, load, and thermal regulation. The characteristic property of an internal current-limiting and thermal-shutdown features make the device being able to be immune to overload. Parameter measurement information is shown below.

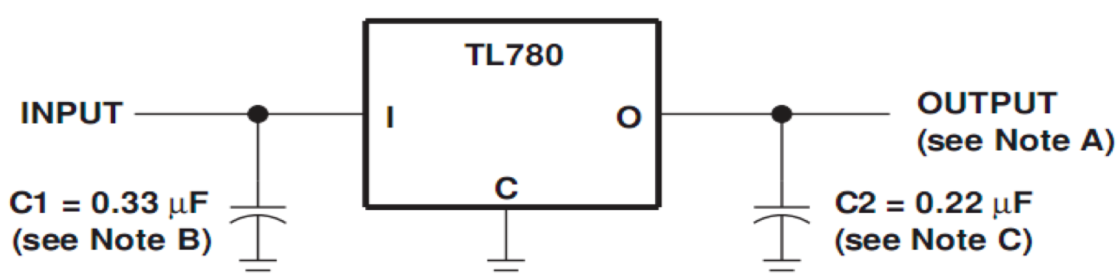


Fig 4.4.5 (1) Parameter Information of TL780
Permission from TEXAS INSTRUMENTS.

From the data sheet provided by Texas Instruments, a permanent damage can happen when the output is being pulled away below ground. The C1 would be required when this regulator has a wide distance from the main power-supply filter. C2 is not needed for stability, but its usage provides an improvement in transient response. However an error could occur when a given input voltage to the designed regulator can collapse at a faster rate than the output voltage. If the derived output voltage is much greater than a given 7 V, the emitter-base junction of the series pass element this could be internal or external can cause a breakdown and be damaged.

4.4.6 LM317

The LM317 has the ability in offering a voltage regulation procedure. In comparison to other voltage regulators. The LM 317 voltage regulators comes in a package that has the ability in regulating an output voltage from a 1.25V DC to 37V DC range, the regulator is adjustable. This is done by introducing a reference voltage to the present third lead. The regulator has the ability to produce a maximum output current value of 1.5A, which definitely should be enough for the components function ability. Few of the features are as follows:

Features

-Output current in excess of 1.5A
-Output-Adjustable between 1.2V and 37V
-Internal Short-Circuit Current Limiting or Output is short-circuit protected
-Internal Thermal Overload Protection or Current limit constant with temperature
-Output-Transistor Safe Operating Area Compensation
-TO-220 Package like 2SC1061 transistors.
-There are 1% output voltage Durability
-There are max. 0.01%/V line regulation (LM317), and 0.3% load regulation (LM117)
-There are 80 dB ripple rejection

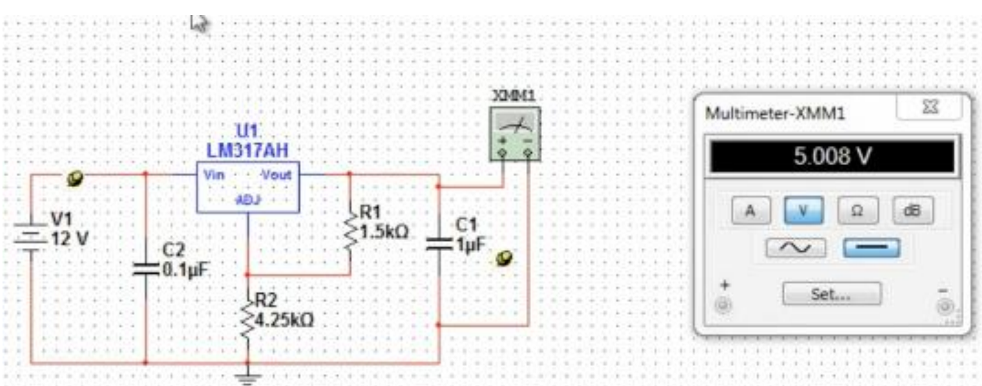


Fig 4.4.6 (1) LM317 Circuit Diagram.

The LM 317 voltage regulator has many advantages not limiting to its cheaper price. The regulator's price varies around two dollars but its basic advantage refers to its compatibility and complete packaging. This device is designed in such a way to be able to receive input voltages and efficiently produce a reliable output stage. A heat sink may be required sometimes depending on the required power needed for the circuit to be designed. The disadvantages to the LM 317 are relatively small. The regulator would require replacing its whole unit if any failure occurs as such careful analysis should be made when using it because of its cheaper price, it's more affordable but time being wasted will be an issue.

The regulator is made of 3 pins, there is the adjustable input where provision from the potentiometer to make the necessary adjustment to the output voltage is made. Next is where the variable voltage can be made, this is linked to the second pin and it's usually a capacitor. The third pin is known as an input pin which is provided from a bridge rectifier or any available constant dc source. However we are considering testing both the Lm 317 and lm 338 for professional knowledge but we might probably use the LM 338 regulator.

4.4.7 Simple Switcher Power Module.

A more ideal approach will be to use a switching mode DC to DC converter or mostly referred as a switched-mode power supply. To prevent the consistent dissipating unused power as heat, a dc converter is designed to have a semiconductor devices in order to allow the provision of the supply input power to deliver the required power at specified timing, the inductor and capacitor components are charged in this cycle. D referred as a duty cycle, is known as the ratio of on time of the semiconductor to the desired switched period. The changed duty cycle will enable the convertor in using required minimum input power in order to reach the intended output power.

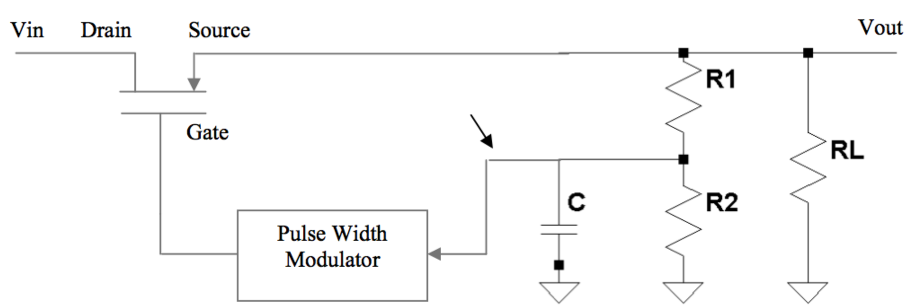
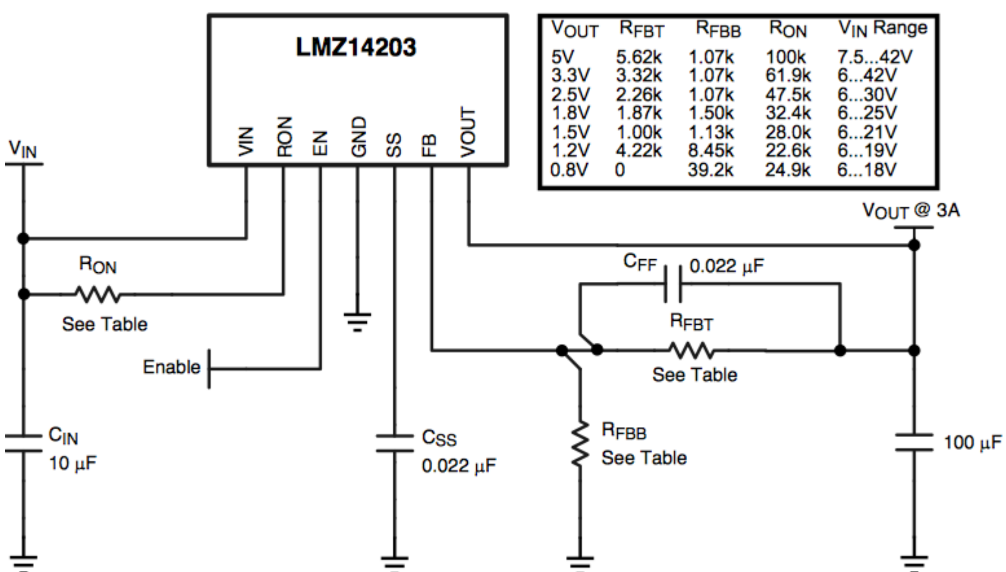


Fig 4.4.7 (1) DC to DC Converter
From UCF Electronics Laboratory Manual

The switching mode converter can be made adjustable by altering the PWM on the microcontroller from the duty cycle of the regulator. Another good reason for us to consider in using the switching mode converter will be due to the fact that it has a wider complexities. This, allowing for the most control of the output voltage and the relative current, as compared to a common linear regulator. For the out power to be changed is a lot of task involved on a linear regulator due to the abundant cases involved with linear regulators being a pain especially when trying to make hardware adjustments. The commonly known two switched-mode converting approach involves the buck configurations which are made in order to decrease the voltage and increasing the current. The buck-boost can have a voltage gain less than, equal to, or greater than one.

Considering critical key features including an output power of 18W and current being delivered at an output of 3A, the LMZ14203 is being considered. This step down converter is designed in a way to have 3 external resistors in place and a four external capacitor arrangement make up the power solution a success. The regulator has the ability in operating on input voltages ranging from a value of 6V to 24V and outputting smaller voltages ranging from 0.8V to 6V and a rated ninety percent efficiency from the data provided by Texas Instruments. The he LMZ14203 converter can provide a delivering voltage up to 5V from a designed 12V input and is generally considered to be a relatively good step down converter.



V_{OUT}	R_{FBT}	R_{FBB}	R_{ON}	V_{IN} Range
5V	5.62k	1.07k	100k	7.5...42V
3.3V	3.32k	1.07k	61.9k	6...42V
2.5V	2.26k	1.07k	47.5k	6...30V
1.8V	1.87k	1.50k	32.4k	6...25V
1.5V	1.00k	1.13k	28.0k	6...21V
1.2V	4.22k	8.45k	22.6k	6...19V
0.8V	0	39.2k	24.9k	6...18V

Fig 4.4.7 (2) Schematic of LMZ14203
Permission from Texas Instruments.

Pre-Built 5Vdc Regulator Circuit Board is another approach for our voltage regulations would be using a pre-built voltage regulation. This comes out as a powerful solution and is efficiently designed and well fabricated voltage regulating board. While there are many such of boards on the market, a very favorable one is the Output 12 and -5 Voltage which comes out as a dual DC Regulating Kit. The board has the capability of accepting an input of 14 to 24 Vac or (+/-) 14 to 24 Vdc input. This finally then outputs the two needed regulated voltages of 12 Vdc and -5 Vdc. This kit is produced by EID Corporation.

The main advantage from this kind of kit from Eilrich industrial developmental board is that it has a well-designed package. The only needed requirement would be to make a connection only to the main circuit with no extra needed design configurations. This could be more helpful in a timely mannered project. Another advantage is its ability in accepting AC voltages. Although not a needed requirement, it could be more useful in a future application. However, we mentioned this in our project because of ifs varying voltages whilst being able to get a constant output supply.

This kit provides a wonderful circuitry mechanisms since it has the ability in regulating the output voltage and not only making adjustments relatively proportional to the input. However as there are minor drawbacks in every designed unit out there, this kit has its own disadvantage as well. This is mainly the cost which makes them pricier to the entire project since as said earlier, our aim is design a lesser or a more affordable 3D scanner. This makes it a huge draw back. Another approach will be to acquire a raw board and installing the components. This is still more expensive on average. For physical component requirements, detailed description could not be obtained from manufacturer's web site but this seems more ideal in a bigger project like. Given the size restrictions of our project, it would not be a wise use of space to dedicate so much space to a single function of a single subsystem.

4.4.8 LMR

LMR10510 and LMR10515 are regulators designed to be used as a step down. They are interesting regulators since they have the ability to drive 1A and 1.5A respective loads and designed with an internal 130mΩ PMOS switch. Switching frequency is internally set to 1.6 or 3.0MHz, this enables them to be used effectively as applied smaller surface mounting inductor and chip capacitor.

The LMR'S regulators can allow as low as 30ns for extremely high frequency conversions over the entire 3V to 5.5V input operating range to bring it down to the minimum outputting voltage of 0.6V. Although we plan on not using this regulator, we thought it was good to have knowledge of this existing regulator. Hopefully in another design we can use them to see how effective they can be.

5.0 Design

Our project was based on the MakerBot 3D scanner (Digitizer) that we mentioned in ‘Similar proposals and Projects’ section. The scanning setup was composed of five primary modules: Motor Control, Laser and Camera system, Power system, Processing Unit, and PC Computer. The setup differed in one point from the MakerBot scanner is whereas the MakerBot scanner as two laser and a camera, where we only implemented only one laser and one camera to capture information. Since we only used one laser our way of capturing data was very different, it was not robust as MakerBot scanner. Our goal was to make cheaper 3D scanner. So we went ahead to make sure our scanner being used was made of few components as possible. We then used the Arduino as processing median between camera and computer to transfer data. And control position of object with stepper motor and few additional materials. It is also used to turn laser on and off. We needed to turn laser on and off because we used the filtering (see section Laser Triangulation under Data Capture.) All the calculation was made possible in the computer using open source software, which is also faster as well. Since 3D calculation does need lot computational power. The goal afterwards was to implement the role of the Arduino UNO into our own embedded system design.

The object was placed on a platform (turntable) which was mounted on a stepper motor which allowed it to turn as needed. We used the stepper motor because it made steps as small as 0.045, making a better resolution. The processing module was taking the images from the camera and processed them in order to produce a set of distance using triangulation method. The processing was done on a computer in order to figure out the distance from the laser to the object’s surface. The laser and camera was then separated so that the laser light line and camera rays did not meet at small incidence angles. Otherwise the triangulation method we used would have generated large errors in calculation.

Also if the angle was made to be too wide on the surface of the object, it was impossible to calculate any data, because camera won’t be able to detect any light from laser. To put it simply two planes formed a right angle. The camera was well set off above. This ensured that the laser line was properly deflecting in the same way, rotating the platform too fast created relatively large gaps on the scan. Since we were using stepper motor which was controlling by specifying a number of steps that in turn turned, where servo motor was controlled by specifying a position, the current potion rotation of the motors was not known immediately. Furthermore, each time we made a8.0 Conclusion

3D Scanners have been around for quite a few years and they still continue to be one of the most widely used devices for subsequent 3D printing. We did not get into details about 3D printing. However, it was something that we considered in the future for future projects. Since we started this research, we have learned a lot about this technology which makes 3D modeling and rendering of object relatively easy and we will continue to strive for perfection as we go along this project. There are many 3D Scanning Devices out there in the market. The whole purpose of this project was to make one that was affordable, effective, with high performance, cheap, portable, not so heavy, and that it can get the job done meaning to scan and object in three dimensions, model it, and rendered it in a data representation software.

We have dealt and informed ourselves with several previous works in this matter and we have reached to the fact that to make an affordable 3D Scanning Device we had to constraint ourselves into making sure that the components that we bought will have all the reference material as well as the desired specifications to actually implement our design. We did not lose time dealing with wrong components since we did not have that much time to accomplish our goal which was building our 3D Scanning device, coding our microcontrollers, testing, debugging, assembling, and presenting it. Basically, every 3D Scanner out there is expensive, big, not so portable, and they require more work and effort put to it. That was the main reason why we chose this project as our final thesis. Also, every single one of the team members had skills that must be polished throughout this thesis. However, this does not mean that other team mates couldn't comment about it. We were just trying to make the best possible within our reach 3D Scanning Device out there.

We did look at a forward to start building our first prototype, and started coding along the way this semester; therefore, we designed something good and probably one of a kind 3D Scanner since we implemented with a series of algorithms developed by us and that has taken a very important role in the final product. Also, there was still a lot to play around with, especially with lasers and microcontroller, our design was not definitive, it did have minor changes when we started building it.

We have learnt a lot over the course of this semester, we dedicated so much effort in designing and writing documentation last semester. This was a team group project, everybody worked in their area, made research, came out with new ideas. One final thought to every adventurer out there that wants to start a project: Team work was absolutely necessary in these types of projects, without that, no project could be completed or even started. It required a lot of sacrifice to make time to work on it, patience, and most important discipline in completing assignments and deadlines. While rotating the motor, we took notes of that in the microcontroller's memory, transmitted it back to the processing unit. We then designed our power supply which supplied enough rated power to our component the motor. We then when connected to the PC with a USB cable so there weren't a necessity or need to power our logic side of the project.

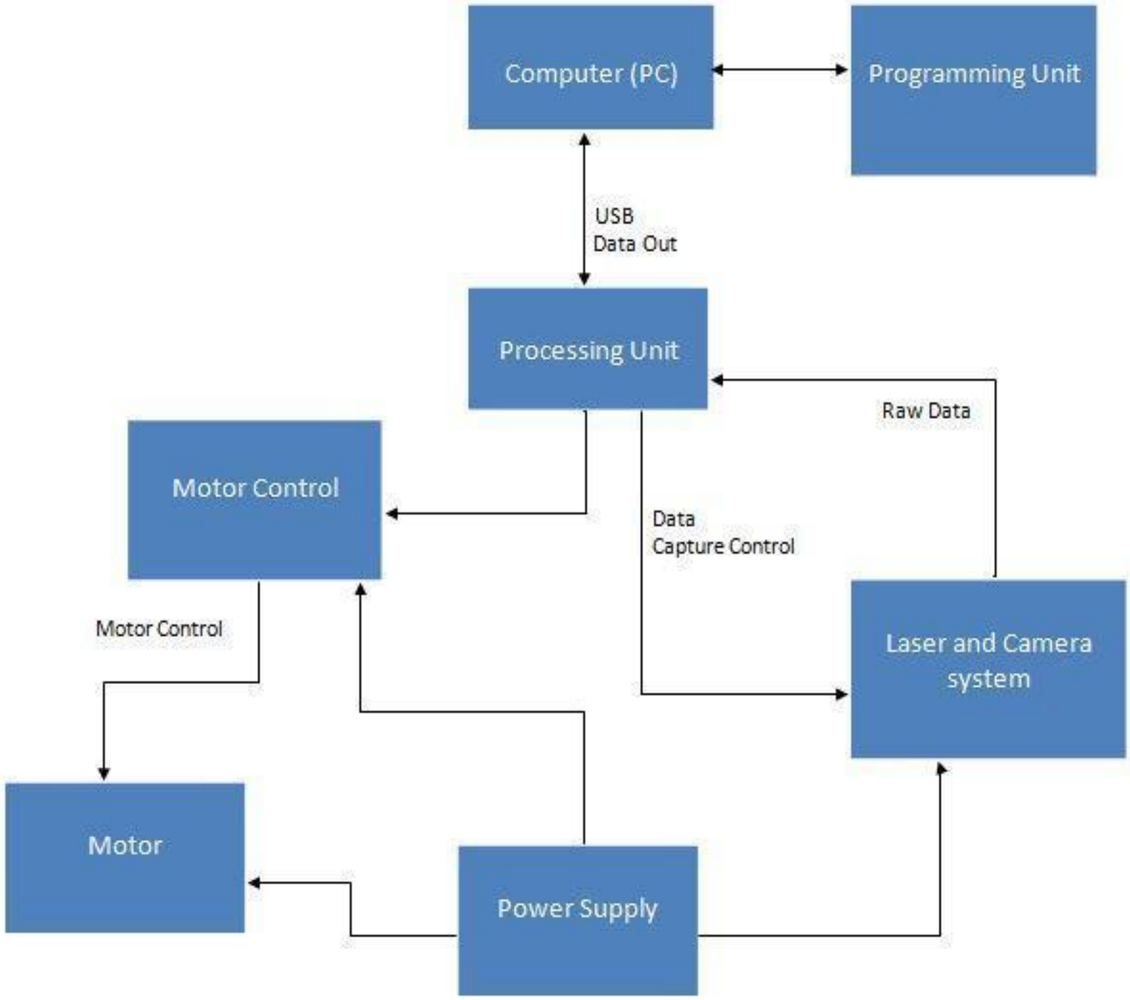


Fig 5.0 (1) System Block Diagram

The surrounding light had great influence on the scan so we either turned down the room light or we implanted using a box around the scanner. If the surrounding was too dark, then camera was not working properly (want to capture image with minimal gain). The controller module consisted of control logic and a USB to UART converter in order to be used as the interface between the controller and the computer. In order to control all of the different components of our scanner system, USB and UART protocols were used. UART is serial protocol which was used to define our control signals. When idle, the data line was high. 8bit messages were transmitted beginning with a start it with zero and ending with a stop bit one. The UART signal was totally inverted and supplied to the motor controller. The motor was step once when a 0x0 byte message or signal low pulse was sent on the UART line. Same way, the UART signals was triggered by the camera to capture image and data was stored on the computer for processing.

For simplicity purposes, we finally chose to integrate together the processing unit and the programming unit into one. That way, we kept the communication the same but the role of the microcontroller became just physical, which is to rotate the scanned object following the computer's orders, as well as sending back information through UART to inform the program whenever an action is done.

Once a full scan was completed, the data stored on the computer was then processed in Meshlab or other open source software we used. The processing algorithm was processed by the camera data to generate a 3D representation of the object. For our project since the laser was fixed and not handled by user. It did have some downside depending on the objects shape, some parts or the surface was never illuminated, and these parts were appearing as holes in the point cloud. Scanning an object with multiple different laser positions minimized the problem from happening. But the cost went up. So we did not implement multiple lasers. Also the turn table only rotated in one axis, it was not possible to scan the bottom of the objects, nor the inside, or the top of the object. We then made holes in the points cloud. Some of the other problems we did face were that color and reflective surface of the object did not have impact on the scanning results, some colors absorbed the laser light (see section Structured light under Laser vs. Structured light).

5.1 Laser Triangulation

For 3D depth recovery, we used line-ray triangulation as explained in Laser Triangulation under Data Capture section. Since we already knew the position of the hardware (camera, laser and platform) we simplified the model without changing the problem (really small precision loss). Since we were doing math in ideal situation we assumed some stuff like laser line was perfectly vertical to object in center. And the camera is pointing perfectly straight and perpendicular to the background. It did not always possible to achieve ideal condition but error introduced was minor.

Since we were fixing position of the camera and the laser we knew the measurement. Also the laser line hitting the background was known. So we knew original position of the laser and how far it was. Then we placed a random chosen object on turn table then the positions of the reflected points on the object surface transformed from the image coordinate system to distance. Then we did calculation one by one, this simplified the problem 3D problem to 2D line intersection problem.

Using two know point's p1 and p2, we created line. y=a*x+b. and point represented as p=(x,y). Then we figured out equation for the line. We then solved for b. At this point we had the complete equation for line 1. We used same method to find line 2. Knowing the equation for line 1 and line 2 we then calculated their intersection.

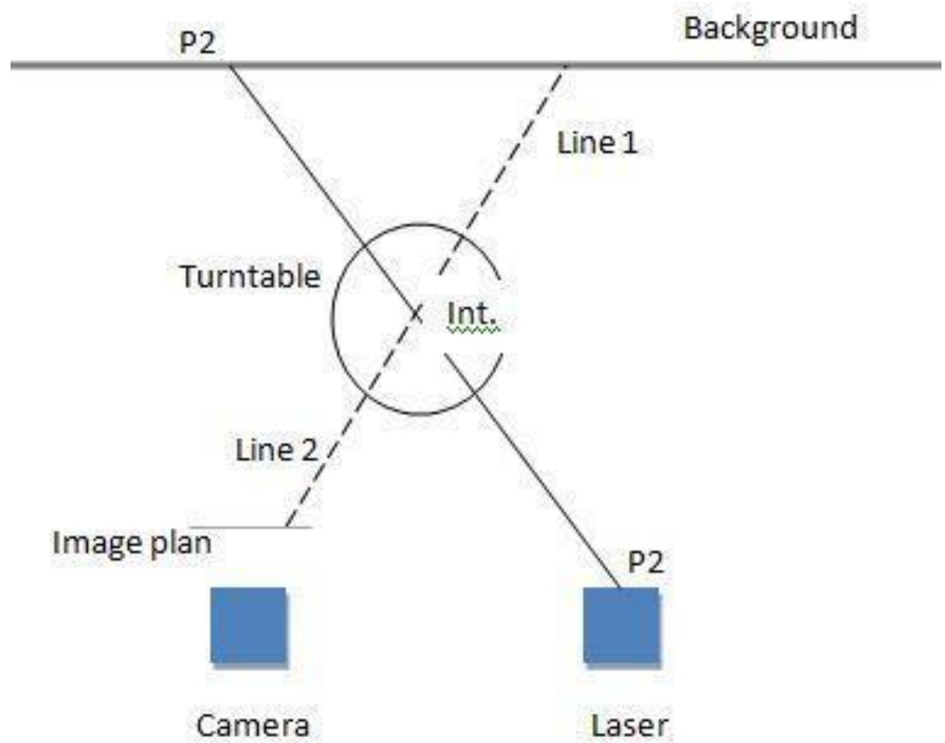


Fig 5.1 (1) Laser Triangulation Draw A

y = a*x+b, p=(xp, yp), line 1 = a1*x+b1, a1=Δy/Δx. , b1=yp1-a1*xp1

line 1=line2, a1*x3+b1=a2*x3+b2, x3= Δb/Δa

y3=a1*x3+b1

Now the depth in real world coordinated of the scanned point was known, we simply scaled the height which was in image plane coordinates to get the world coordinates which gave us 3D representation of the scanned points. We needed to do this every time we rotated the object. And every picture we took. But we still needed to keep how much we turned the object so then we had to rotate the image plane manually to get the right representation.

In this setting we implemented with red line laser. This was then be projected at a background or back wall. When an object was placed on turntable 0.5 meter away, the vertical laser line was appearing as offset from the reference taken from the back wall. As show in programming section, we needed to know some basic information to solve for the distance between laser and object surface. And generally not all cameras were well documented. We were unable to find the size of the camera sensor which was required to determine the size of the object as seen by camera or the focal length of lens.

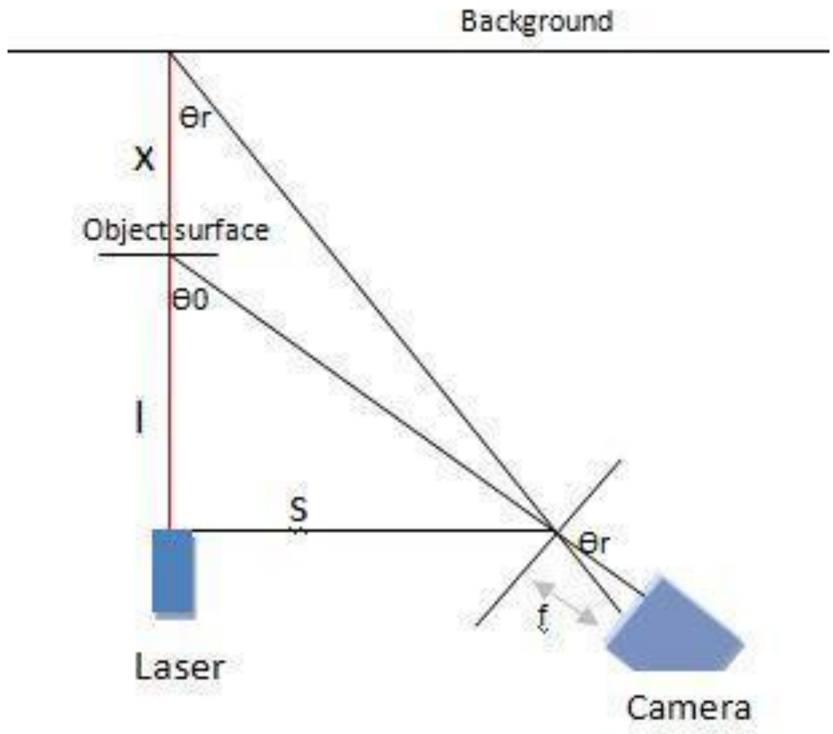


Fig 5.1 (2) Laser Triangulation Draw B

The actual calculation for this method was not linear as shownbelow; it may be roughly be approximated using simple trigonometry.
Variable definitions **:** Ɵr – angle between laser and camera, Ɵs – angle between camera center and projection, Ɵ0 – angle between laser and camera at object surface, f- focal length of camera lens, d- distance between camera center and projection, l= distance between laser and object surface. Since we were fixing the position of camera and laser we already knew the value for s, f, Ɵr, and find l or x will give us real world coordination from image plane coordination. The calculation for the distance between the lasers and object surface,

Ɵ0=Ɵr+Ɵs, Ɵs=inv tan (d/f), l = s/tan (Ɵ0), l=s/tan (Ɵr + Ɵs)

5.2 Laser and Camera Module

The Laser and Camera module produced the data which was to be processed. This module was implemented with a 25 mW 650nm red line laser. The vertical line was about 25cm in length; it was then projected at a blank background. When an object was initially placed on the turn table 0.5 meter away, the vertical laser line was appearing as offset from the reference taken from the background. This image was then captured by a sensor which was then relayed as the data to the processing module.

We were using Arduino Uno; mainly it did serve as transferring image data from camera to computer via USB cable quick as it can. The more samples we choose, the better the resolution of object scan was and the less chance there was for leaving "holes" in middle of point cloud beside the bottom and top. But image file was always big so this was made as our bottleneck for the project.

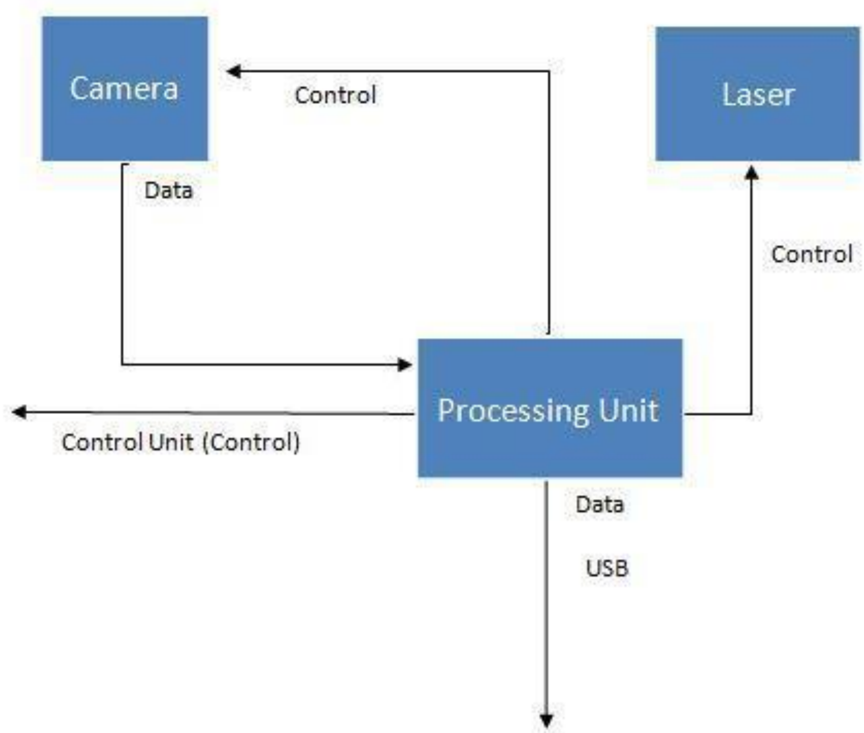


Fig 5.2 (1) Block Diagram of Laser and Camera Module

But we did not do any calculation in Arduino, and the only other function was giving signal to motor control to rotate the platform, for it going to work decently. We chose Arduino Uno because of cost, and simplicity. We could have always added on to with Arduino shield without throwing away the old component. The physical setup was such that the camera was offset by 15 cm from the laser, and that was accounted for when programming the processing unit to calculate measurements through triangulation. Also on the next page there is a flow chart on how to program the processing unit.

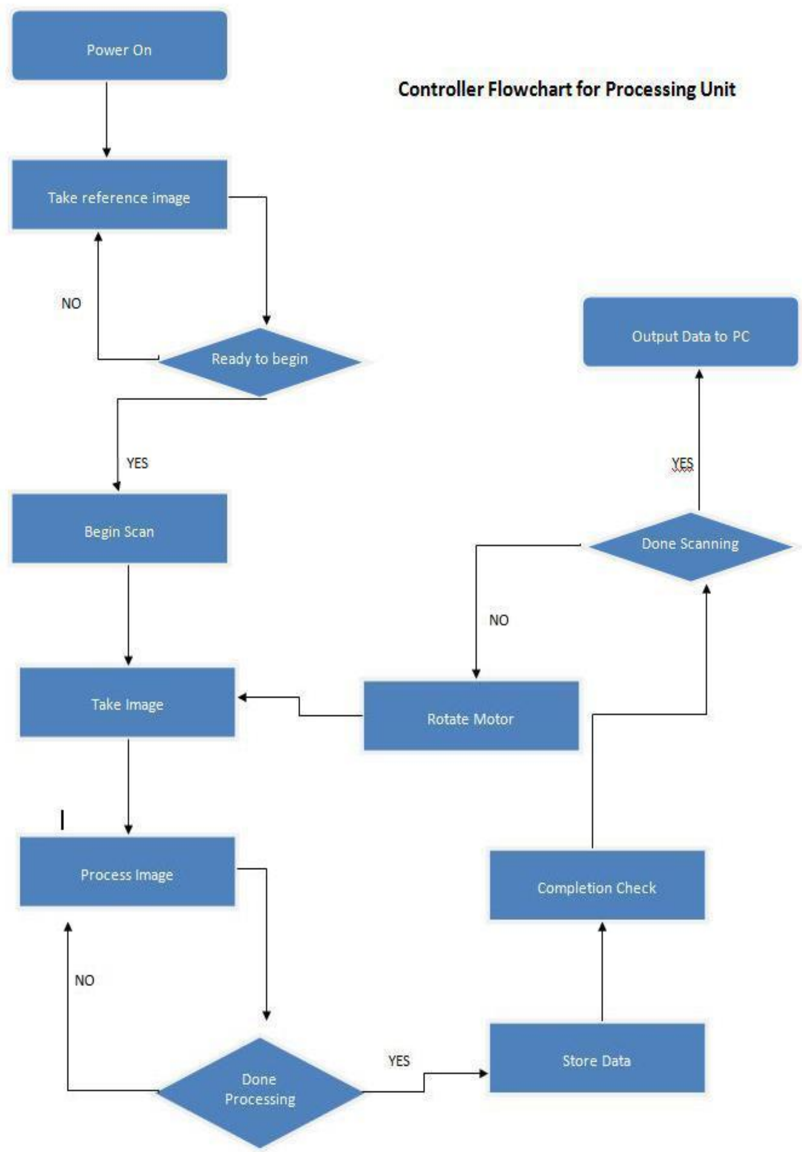


Fig 5.2 (2) Block Diagram of Processing Unit

5.3 Motor System.

Our main mechanical function of the project was rotating the disk on which the scanned object resided. Every rotation was made in the order of a few degrees each time, so we took snap shot with the triangulation process which was ready to get to form the point cloud at the end of it.
After evaluating the pros and cons of Servo motors, stepper motors and DC motors, we came to the conclusion that Stepper motors were the best suitable for our projects for a few main reasons:

- We needed accuracy. Until we reached a 360 degrees full rotation to scan every side of the desired object, we needed a steady and consistent rhythm of turn. For example, when we tried to take 180 snaps for our point cloud, that meant:
  360 degrees / 180 = 2 Degrees, for every rotation. As we imagined, consistency was absolutely crucial, so we did a mathematically useful approach that had solid and reliable data, we were able to use in some aspects of our project. When our full rotation was off by a few degrees, or not complete when the snaps were done, that did give us problems in a lot of ways, especially for the image processing that did not give us an accurate representation of the scanned object. The same went for rotation, when it was different than another, like a 1.8 degree turn followed by a 2.2 degree turn.

  Even though they lead us to an overall desired position, the data taken from it was erroneous putting in jeopardy the reliability of our project. From the different stepper motors found online, and seeing what the nominal minimum step for each model was, a 1.8 degree step was the kind that was among the shortest and most convenient for us, lots of those were available. That also meant we could drive the motor to make half steps or quarter steps needed to be even a shorter rotation for more operating angles.

  Servo motors are built more for a performance utility, like a fast spinning wheel of a robotic arm, whether accuracy and precision of more in the steppers domain, making stepper motors the natural choice needed for this requirement.

- The second point had to do with torque. While some servo motors had high torque specifications, it was more common for stepper motors to have low speed, high torque performance. DC motors did have very high torque as well, but their difficulty to power up for this specification did not make them a suitable choice. For our purposes, and object of undetermined weight will be scanned. It was to be light weight paper object, or a mechanical part made of metal or other heavy material. And for that we had to be prepared to carry and rotate as heavy of an object as we did scan, that fitted on top of our disk turned by the motor. Torque, the product of inertia and angular momentum, was synonym of direct, raw mechanical force. It was then an important requirement that made once again the stepper motor the most logical choice when compared to Servos and DC motor alternatives.

- Another thing was that stepper motors had a continuous functioning. In fact, they were controlled for a step-by-step behavior. With no feedback or memory, all they needed was the required pulse to move with the designed step, regardless of what happened before that step or the rest of the program, if applicable. They were able rotate or make a full 360 degrees (a must for our project) and kept going when needed to, and that seemed to be a perfectly fine situation for our needs.

  Although usually, they were supposed to reach a full rotation with the exact number of steps as predicted/specified, but sometimes a rotation could not be performed, according to some research undertaken, because there was something holding the shaft from turning or blocking it until it couldn't move. In our case, that might have happened if the object to scan was too heavy, and this was another reason why we needed to pick a motor with sufficient torque, because besides the visual and other sub-systems of our project, we couldn't have a way of knowing if the motor actually turned or not (we provided a means of always finding a remedy for that in the processing part of the project, or the "brains").

After comparison of the few motor researched, we concluded that it was safe to choose the NEMA-17 17HS15-0404S for our design. Of all the stepper motors its voltage category (9 Volts DC), it was the most powerful stepper motor with 56 oz.in, the others barely going the above 30 oz.in line. With a 0.4A rating, it was work working simply and in a straight forward manner with our stepper motor driver of choice, Texas Instruments' SN754410 that provided up to 1A current per phase in a simple and efficient lay out, adapted to our needs. At less than half the price of the higher power version we analyzed, the NEMA 23, this motor choice was sufficient as it will cut from the final price of our budget-oriented 3D scanner device.

5.3.1 Printed Circuit Board Design.

For an effective and ergonomically way to implement the driving system of the motor, all the parts had to be reliably connected. For that, we then made a custom printed circuit board that we used to go to specially order for the sake of our project. This process was meant to go from selecting all the required components for the project to putting them together in a schematic. This schematic was made to be part of the PCB layout, with every pin connection and component detailed into it. From this schematic, we easily designed our circuit in the prototyping phase so we did a couple of testing of how everything was running. Even though they looked different, for the sake of prototyping, this was made to get identical results as we desired in our final product, which was then on our designed printed circuit board.

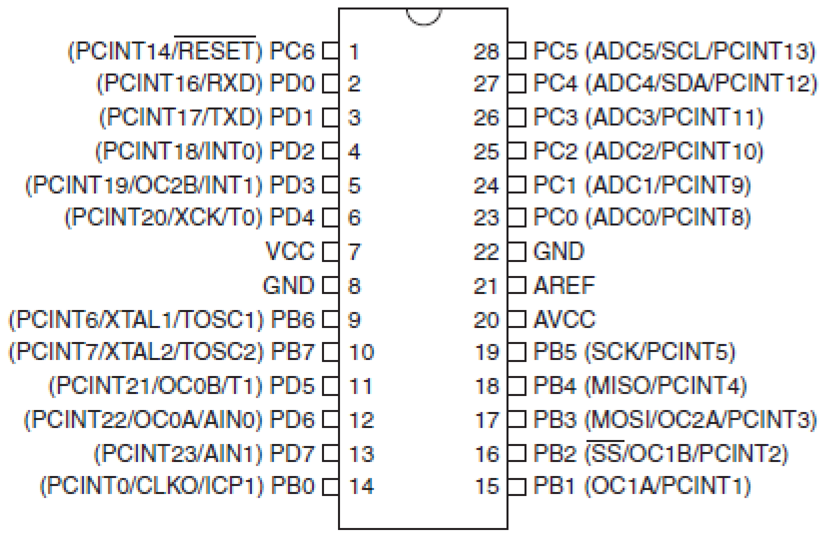


Fig 5.3.1 (1) ATmega328 pin diagram
(Approval from Atmel)

The way we actually implemented control was through our Atmel AVR AtMega328p microcontroller, which was sending bits to the stepper motor's driver in order to rotate the disk in the desired way. The diagram below gives a general idea of the functioning of the motor sub-system; expect we are using an ATmega328 instead of the Attiny45. The reason for that change in design is that we realized that we needed more inputs/output from our microcontroller, including the 4 outputs for the motor, Laser/light activation, as well as serial UART communication with the processing/programming unit We figured out the various components that were going to be on the printed circuit board and it was a success placing all our components on our final schematic. We planned to design it on Eagle, software so that an easy access through the university facilities were made, although similar, free PCB design software, PCB Artist was accessible from 4pcb.com (the manufacturer we chose for the final product) and this was used first, in laying out a preliminary design. After laying down all the requirements, we then made a first schematic to have a final view of the final product.

To implement communication, we chose a common USB to serial (URAT signal) converter, the FT-232RL made by FTDI. The data pins (D+ and D-) on the chip were connected to a mini-USB connector that linked the PCB to the Computer where the program is running. Then Its RX pin was connected to the TX pin in the MCU, and the TX pin was connected it its relative RX pin. That was we assured UART connection between the PC and the microcontroller, which has a function of sending 1 commend byte at a time to tell the MCU what to do to the motor and laser, and receive a byte back after each action is done, so that the program can move on to the next step.

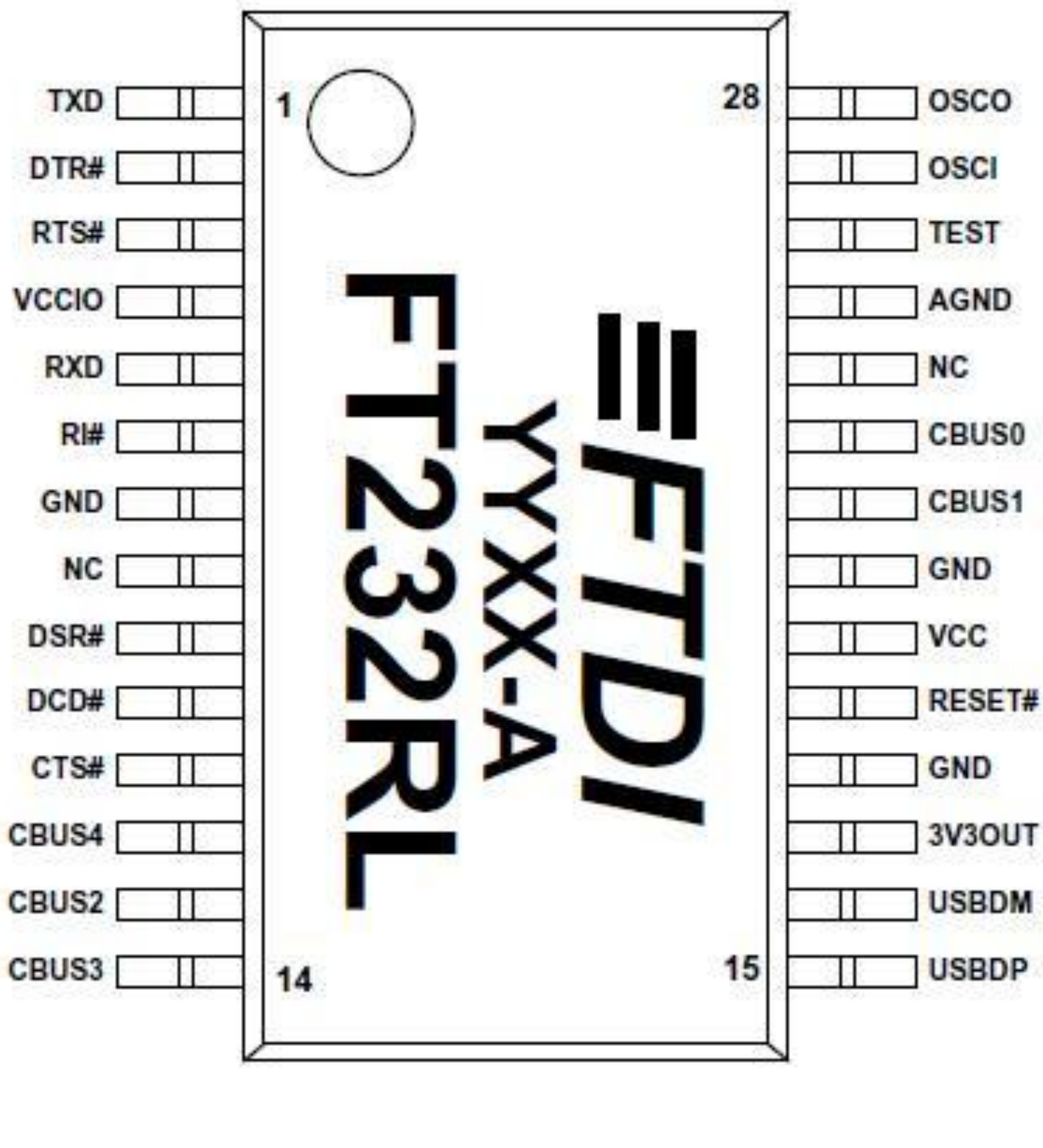


Fig 5.3.1 (2) Serial communication converter chip pin-out diagram (Approval from FTDI)

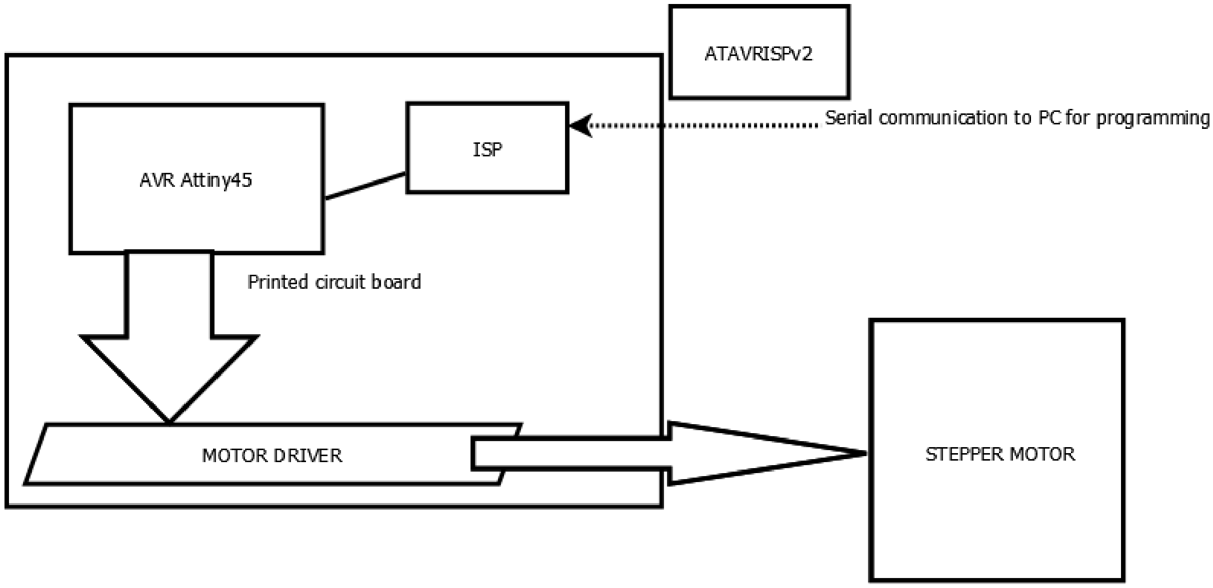


Fig 5.3.1 (3) Motor Control-General Diagram

5.3.2 Motor Driver Implementation

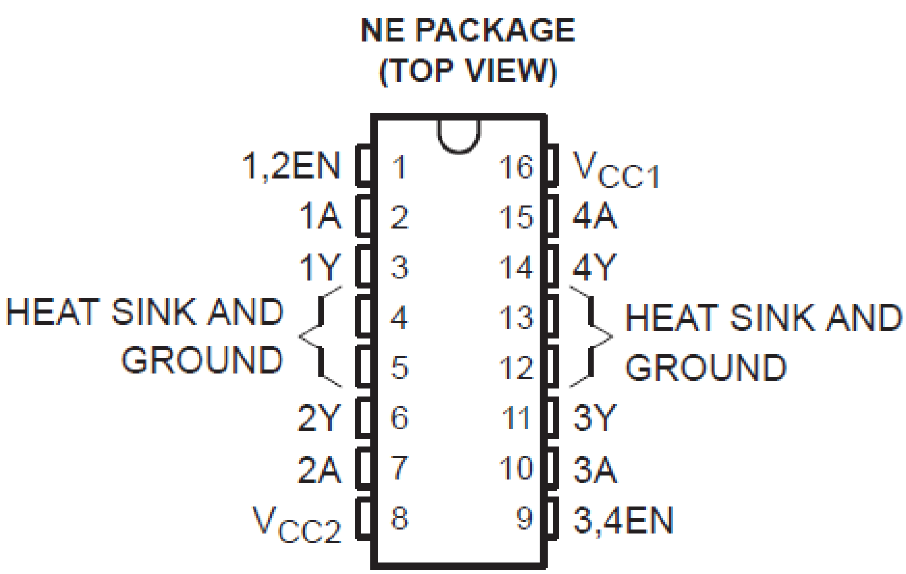


Fig 5.3.2 (1) SN754410 Integrated circuit Pin diagram (Permission from Texas Instruments)

As we can see from the pin diagram, each input, denoted xA, had its corresponding output, denoted xY. Because of the high current flowing through the integrated circuit and the power that was generated when the voltage was high enough, 4 pins were dedicated to be grounded serving as a heat sink. It was keeping the system from overheating, and this was extremely important in this case where the circuit was mainly made of transistors, semiconductor devices in which multiple properties like the breakdown voltage were high depending on the temperature of the materials. Keeping the temperature at reasonable levels permitted the device to function just as the manufacturer had documented it to work, so we relied entirely on it to function correctly in our design.

The pins 1,2EN and 3,4EN of the driver were able to enable the corresponding pair of pins, thus having it set on HIGH all the time did save us something else to control, with more output pins and more programming to do. It was just a shortcut to further simplify out design and avoid possible complications. To do this we we had them wired directly to VCC (+5V) so that it worked all the time that the device was powered and connected to the logic inputs. The following table illustrates the situation, as well as the logic diagram that showed us how the IC's system worked. It's in a double inverting gate form, in which the outputs of 1,2EN and 3,4EN controlled each pair of inputs-to-outputs gates. As the manufacturer recommended for most similar application, we then connected them to VCC throughout a 10k ohm resistor.

FUNCTION TABLE
(each driver)

INPUTS†		OUTPUT
A	EN	Y
H	H	H
L	H	L
X	L	Z

H = high-level, L = low-level
X = irrelevant
Z = high-impedance (off)

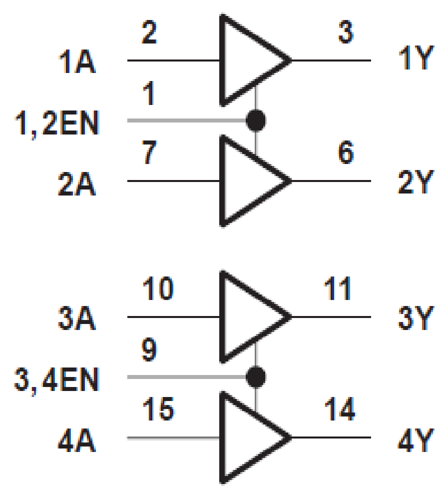


Fig 5.3.2 (2) Function Table
(Permission from Texas Instruments)

5.3.3 Interface Serial Programmer (ISP)

Thanks to Atmel's fine documentation on the subject, we came to the final conclusion that including the programming interface with the final design was the straightest forward, cheapest and most reliable way to implement programming. Whether it would be the prototype stage or the final design, we always had the opportunity to change and improve our code as the development of the project went on.

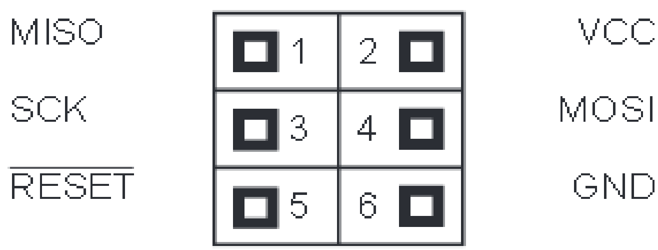


Fig 5.3.3 (1) ISP Diagram

According to the AVR guide, to program the AVR ATtiny when implementing in-circuit programming, we needed to have the RESET pin, or pin 5 on the MCU, on active high. This was to be automatic, as it was designed as a function of the REST pin when HIGH. Once that was done, we then had the device ready to be programmed, as the SCK, MISO and MOSI function of the corresponding pins was to be enabled. The main I/O we were using was trough MISO/MOSI. More resistors were added.

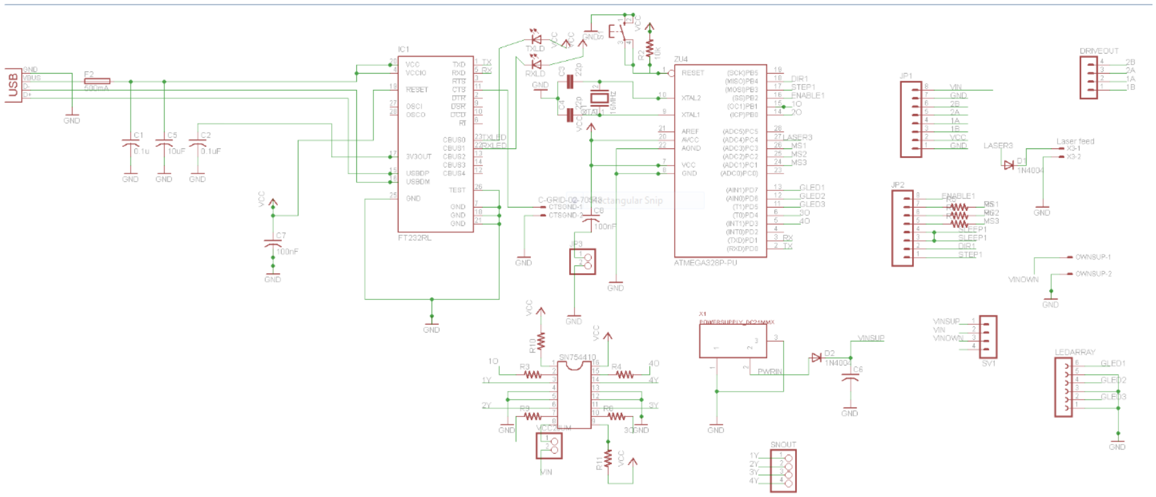


Fig 5.3.3 (2) Final printed Circuit Board Lay Out Schematic

We also gave the needed connectivity to the GND pin with the ground that will go with the 5V power supply. The reference voltage needed to be the same for operating and programming.

After changing the microcontroller model used for our project, the programming way remained exactly the same since it the same manufacturer, so the way we program the ISP is exactly the same with the dedicated MOSI/MISO, SCK and RESET pins in the Atmega328p.

5.3.4 Microcontroller Programming

The way we were looking to program our microcontroller was through the C and AVR assembly language. While assembly was to be very neat and more hands-on the way the microprocessor operated, C being a more classical, well known language was made more efficient and a more straight forward way to implement our algorithm. Besides, whether it wasn't in assembly language or C code, we used the same software tools, part of the same Atmel studio bundled to program. That way we were able to always switch between the two methods when we judged one being more efficient from the other for some reason.

The inputs of the AVR microcontroller consisted of the input communication pin with the image processing unit. This was going to tell us whether we needed an action, or one motor step leading to one snap from the camera. A simple YES or NO requirement was what we were dealing with. Yet a communication system was fundamental and setting it up properly was the only way to insure reliability of our project. This was the deciding factor for the outputs. Most serial communication protocols, like ISC and UART, worked with both transmitting and receiving interfaces. That meant if we got to implement a way of communication, we needed an output pin set up to the receiving pin in the processing unit. The transmitting pin, known as Tx, was driven to the receiving pin on the AT mega (The microcontroller in the Arduino processing unit), named Rx. In the same way, Rx on our MCU was connected to Tx in the processing unit. With this set up, we were then able to send bits of information back to the processing unit. Here, primary information we needed in knowing how many rotational steps of the motor were undertaken. We were able to program it to be able to send this information in real time, in that case we were able to make the "number of steps" variable increment in the program after each rotation, or do it inside our MCU and saved it in the FLASH memory and then sent that number from the Tx pin after the final rotation for the process to go on the next phase. We then added other functions or data to transmit later as prototyping goes on. We insisted on this aspect of the communication because it was proven to be vital to the fidelity of the point cloud we were trying to generate. The latter option was chosen to let the MCU fully operate the controls and the communication.

The way we able to turn the motor one step, was by enabling one of the 4 outputs, or setting it to a high voltage. This went to the motor driver at the corresponding input and consequently turns on the corresponding output directly linked to one of the coils. After that, when we went ahead to make another step of the motor, we activated the next output on the list that was in consequent turn on the corresponding input/output on the driver. For example, let's say we Started with the pin1 output. Once set on HIGH, it went through pin1 input at the stepper motor driver, and activated the first output powering up the first one of four coils.

Once the voltage drop occurred, the magnet on the shaft flipped halfway, making one motor step. Then, when the signal came in that we needed another action, we had to use, in a sequence way, pin 2 output of the microcontroller to enable the second output of the Stepper motor driver wired to the second coil, clockwise. This was made possible for the motor rotating a step a second time. In the same way, the next step was made possible by triggering pin3 which will enable current on the third output of the driver, turning the motor by one step again using the third coil. After doing this a fourth time using output pin4 of the microcontroller and powering up the fourth coil, we did have to go back to the first pin of our device, setting up a loop going through these four outputs or commands, and this was our redundant way of controlling our motor step by step, for a full rotation of the motor, which we went through in the process of our system, we needed this loop 50 times (200/4=50). The following diagram illustrates the main point of this situation.

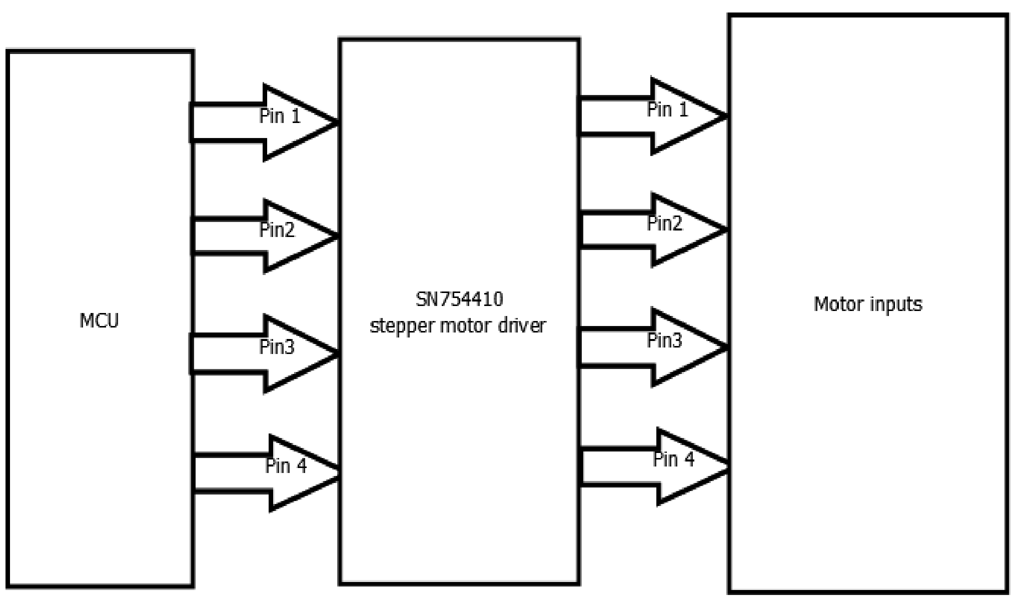


Fig 5.3.4 (1) MCU Stepper Control diagram

5.4 Power Supply Design

The voltage from the AC mains was first of all had to be converted to a DC voltage before it entered the converter. The initial step of the rectifying stage was to be able to filter out the high frequency AC noise by means of a low-pass filtering procedure. A metal oxide which usually has a high resistance at low voltages and a low resistance at high voltages, and this is intended to be used to protect the power supply circuit from the power surges. After this stage, a 60 Hz isolation transformer was providing an electrical isolation from the AC mains.

The output from this transformer was to make an entry to a rectifier bridge which had a filtering capacitor on the output and this provided a change from the single phase AC to a DC waveform. The DC waveform then entered the DC to DC converter, which then finally gave a regulated DC output that was changed in magnitude and corrected by a control circuit if it differs from our desired output.

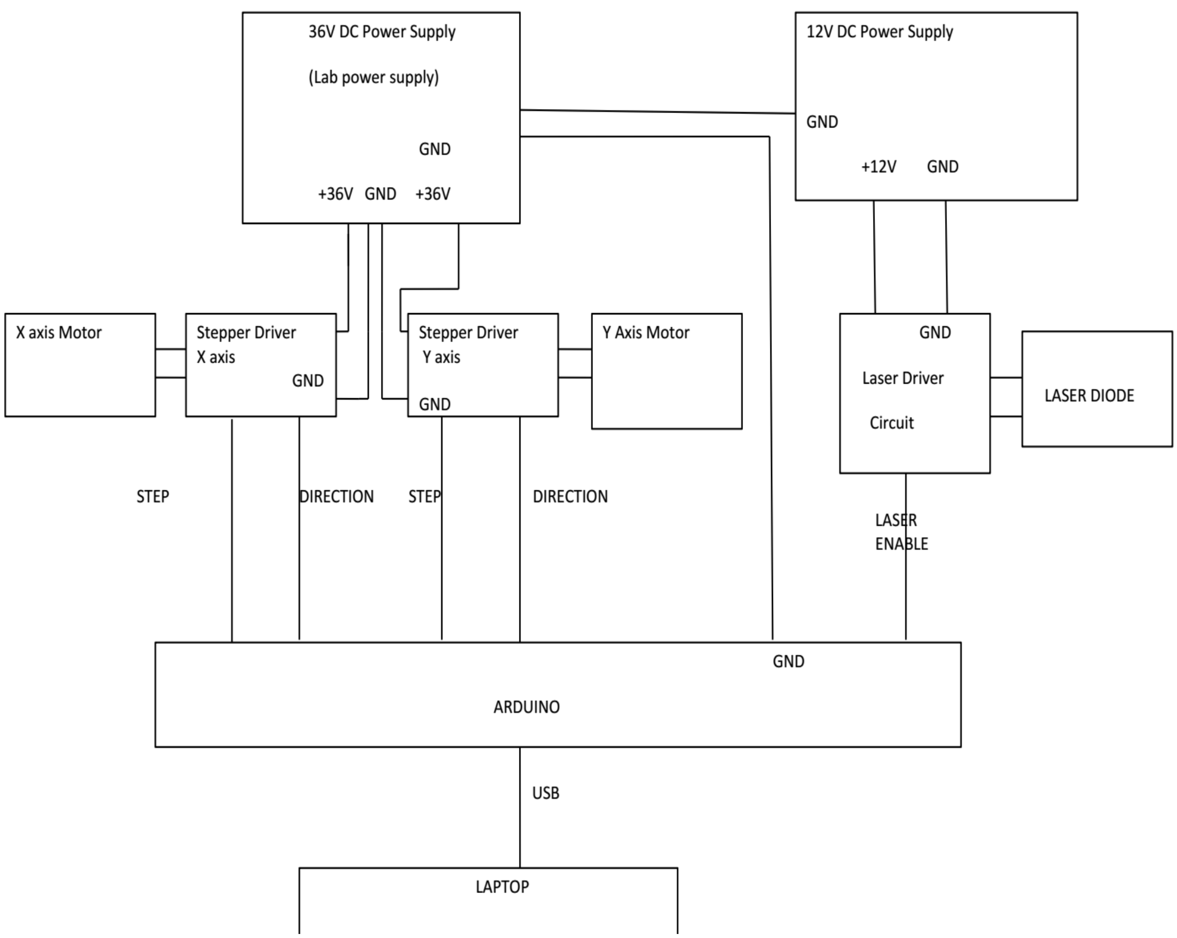


Fig 5.4 Power Delivering Analysis Diagram

5.4.1 Power Control

From the load current passing through circuit resistors was able to generate heat dissipation on the device. An example was from when we used the 12V dc power supply to produce a 9V dc in order to feed our load, this can be able to draw a current of 0.5A under 9V. As discussed above, the LM317 will be adequate for this justification. From the multsim of the 78xx it can be realized that the circuit was able to draw a bit more than 0.5A from the power supply that will result in approximated value of 9V x 0.5A = 6W power delivery. But our load worked will work with 9V and it will be drawing a current of 0.5A that results in 5V x 0.5A = 2.5W significant power being used. The power being delivered and the power being used will result in a difference of about, 6W – 2.5W = 3.5W which was dissipated as heat on our regulator device.

The heat on the regulator had a greater probability of causing a malfunction when the heat being generated finally exceeded the maximum value that the designed package of the device was able to handle. We had two ways or approach in preventing this error from happening. First of all was reducing our input, or we made good use of a heat sink. This heat sink was then connected to the device. From our research, we found out that, sometimes attaching these heat sinks doings may not be possible. This was be verified from our prototype before we applied the usage of an affordable heat sink we acquired from sky craft. We finally decided to use a Step down Voltage Regulator which had a better statistics in terms of efficiency.
The power supply for the entire system was powered both during prototype testing and final presentation. This was powered from the lab's main power supply before being split up to the rest of the components. After we chose the recommended transformer of desirable power ratings, the power distribution analysis to the entire circuit will be designed making the needed voltage was met by each component of the circuit. Even with a voltage drop which was made twice that of a half-wave rectifier, the full-wave rectifier was more efficient because the losses was mainly contributed from passing half of the AC signal.

This circuit designed was simply made up of the various voltage regulator chosen for the project. The main relay was made of a designed circuit which began with a 9 V regulated output voltage.

The Lm 317 voltage regulator was used. By choosing various values for the resistors, different values were achieved. We tested this in our prototype for different voltages. In our case since we only wanted 9v as our output, we chose one resistor of value 750ohms and another resistor chosen was 4.7kilo ohms. Our desired voltage was achieved without any issues. We places couple of capacitors both at the input and out to reduce rippling effect. Our voltage derived was less rippling effect.

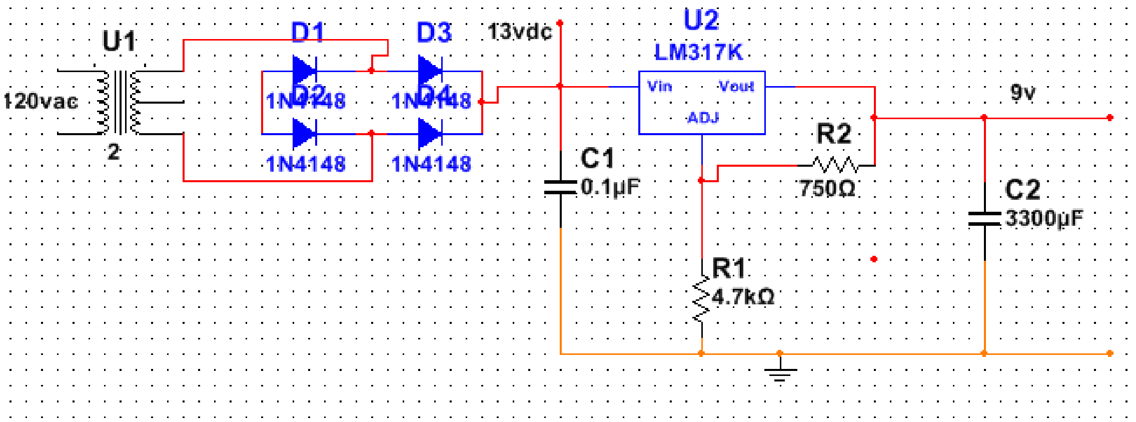


Fig 5.4.1 (1) Circuit Diagram

5.4.2 Regulation

For the required 9V needed to supply sufficient power to the other components needing the 9V, the circuit was designed by getting its supply voltage input from the above circuit of the output of the 12V rectified section. We planned on using a 12V power supply which was then regulated by means of an LM317 regulator from Texas Instruments. All data were derived from Texas Instruments website.

Fig 5.4.2 (1) C.E.S TRANSFORMER

Specifications of transformer:

- Primary Connections: Black Wires: 220V 50/60Hz
- Secondary Connections: Brown Wires: 12 V, Center-tap: White
- Rated Current: 3 amps output.

- 2-5/8"(3-5/8" OA with mount flanges) x 2" x 2-1/4"(H)
- Connection leads are all 6" in length

5.4.3 Rectification

Each pair of diodes was generally made to be in series connections, the anode was connected to cathode. Available diodes on the markets usually had four distinct terminals in order for them to be configured for split supply which is a single-phase, half bridge or three-phase usage.

A very important clue we didn't miss from these rectification processes were losses from the peak input voltage to peak output voltage which was very critical. These losses were generated from the built-in voltage drop across the diodes. Generally 0.7v for the designed p-n junction diode and 0.3 v for the Schottky diodes. This was definitely going to be a problem in the decision to choose a full-wave rectifier since with a half-wave rectifier, the voltage drop was going to be across just one diode but for a full-wave bridge rectifier, the voltage drop was going to be that of two diodes. Even with a voltage drop which was twice that of a half-wave rectifier, the full-wave rectifier was more efficient because the losses made were more mainly contributed from passing half of the AC signal.

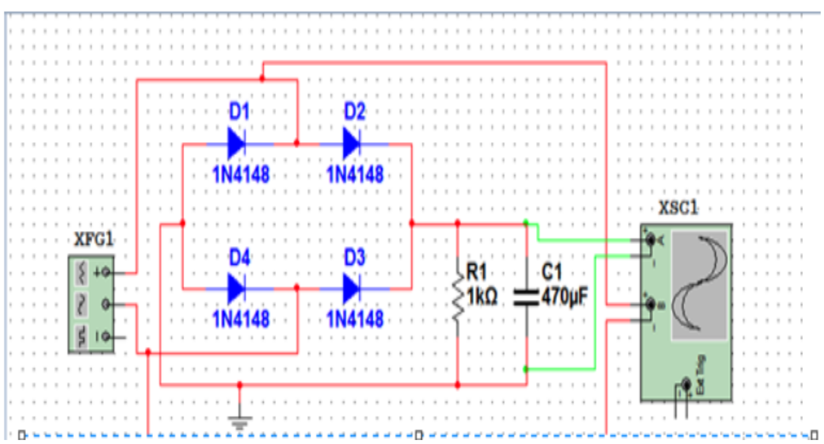


Fig 5.4.3 (1) Rectification Circuit-A

Even though rectification was to be producing a good form of the DC voltage we wanted, this voltage being generated was not going to be a constant DC voltage type. This was due to the fact that there was existence of AC rippling which inhibited the process. This AC ripple was known to be the unwanted residual periodic variation in the DC output which we derived from the AC source. This was basically due to the stage where there was incomplete suppression of the AC waveform. In fixing such a situation, a smoothing or filtering circuit was placed in part of the designed circuit to correct the problem. This smothering effect was simply achieved by means of a capacitor, and this was placed in parallel with the load and the rectifier.

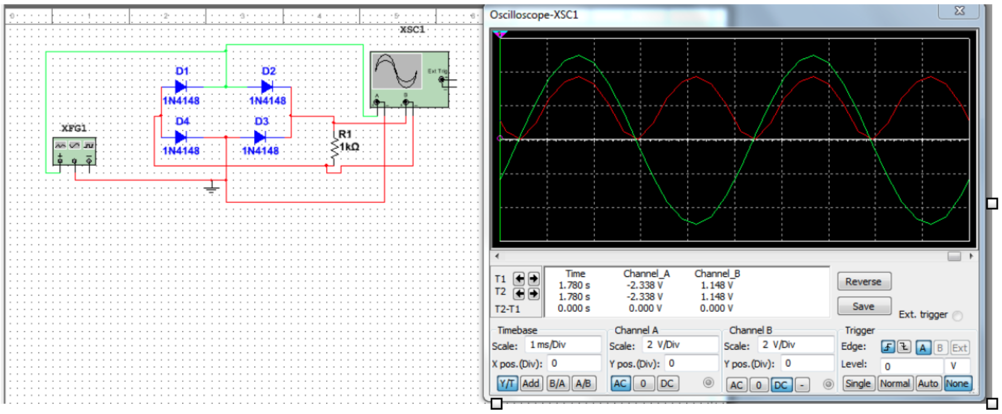


Fig 5.4.3 (2) Rectification Circuit-B

The size of the capacitor chosen indeed affected the ripple effect which was to be smothered out. By using a larger capacitor, the ripple effect was considerably reduced substantially but this was more cost effective and even creates a higher peak current in the supply. Rippling effect was further reduced by placing a capacitor at the input filter. This was made of a capacitor arranged in a parallel arrangement with a rectifier; an inductor was placed in series and a capacitor again in a parallel combination. For the given tolerable ripple effect the capacitor size to be acquired was to be proportional to the load current and was inversely proportional to the supplying frequency with the number of output peaks of the rectifier per input cycle.

In the figure below was the smothering effect achieved from placing the capacitors.

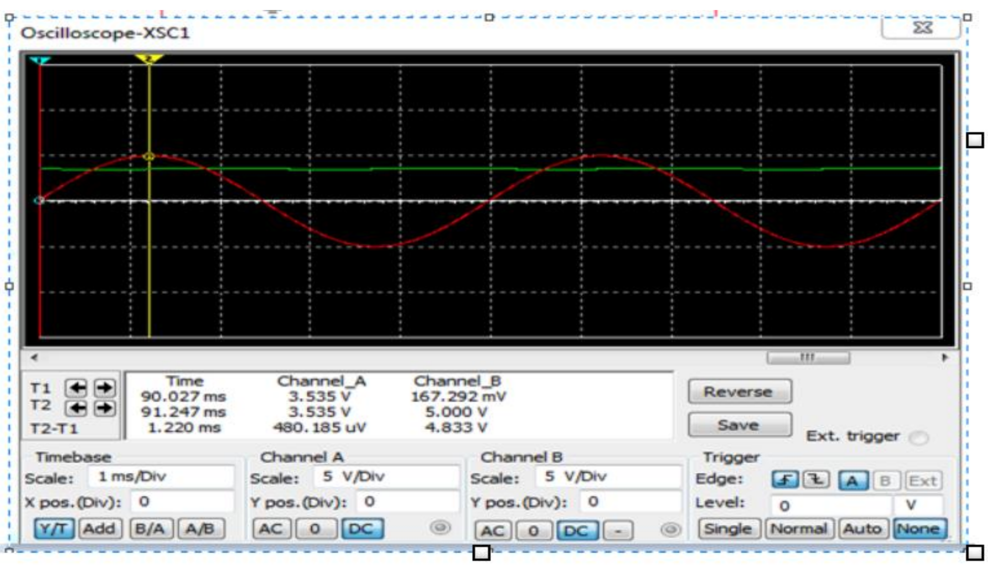


Fig 5.4.3 (3) Rectification Circuit-C

5.4.4 Power Supply Prototype Testing

For supplying the power needed by the motor and the components such as the controller unit, the initial step was to take into consideration what we placed to redesign the board to be used. This was used for prototyping. After the needed components were carefully soldered onto our board, the next approach we did was a definite testing to make sure everything was on point. Testing being made on the boar included making sure the limits was working efficiently as planned and that the emergency shutdown was working as well.

The next step was then incorporating the rectifiers and the microcontrollers to make sure the motors were moving the way we desired. Including the angles that were needed to be able to have accurate data. Making sure adequate and appropriate rated voltages were supplied and avoidance of short circuitry were some of the basic testing evaluations for the entire components. Using available meter readings from the lab, the resistance was verified from the entire designed power rail to the ground. This was done to in order to ensure that the value of the resistance was greater than zero ohms. After making a successful completion of these testing procedures, a careful look for possible shortages which was a common problem to be made in the lab was critically checked for. Any possible broken pieces were checked as well .In case that all the solder joints were fully utilized, a thorough further checking was made.

This was done to check for possible manufacturer defects which could have occurred. All these checks were to provide a verification that all soldered components including capacitors and resistors were matching those to the expected and designed values for our board.

5.4.5 Power Supply Specification

The power regulators we chose for our circuit board design was made small as possible, and quite a number of them were made as a through-hole based solutions which then enhanced a rapid prototyping. Making use of a breadboard alone, and a couple of through-hole resistors and capacitors, it was made possible to be able to employ each circuit designed and be able to test the actual output voltages for significant tolerances and noise by means of an oscilloscope.
We then made a testing on a switching based circuit in order to verify if a component choice, like a resistor value, was going to need a modification in order to obtain optimum output results.

Obviously our design was to be made in such a way that there was going to be various or different components of our system which was to be made of different component power requirements. At this step we designed some DC/DC power conversions in order to have a successful outcome. From efficient power delivery from our relay, the motor driver and the control logic was designed to each have a separate power source since they had a different component parts with different rated voltage inputs. We then obtained a 9V power supply from the wall outlet, and our preferred linear voltage regulator in order to further step the 12V down to the required 9V required. We were able to obtain various information on how the some of the most commonly used voltage regulators work. From the various information we derived, we are considering using the LM317. These were chosen for our step down analysis. They are widely used regulators based on the research we made.

The method we used used to power our laser diode was to be reliable, precise, efficient, and have effectively low dissipation of heat. The laser diode was required to have a constant source of power that would have a precise value to have a smaller voltage and a current ripple from the power output conversion. When the input power changed,, this affectedthe diode functionality and even made the diode being damaged entirely. The ideal power supply we were looking at was required to have a rated input of 120 V AC at 60 Hz frequency, from the AC mains and being able to provide a desirable output of 0 to 5 V DC at 60 A to the laser diode we were using. The output voltage ripple from our design, was to be less than a one millivolt. We are designing and simulating the voltage for the required values to be determined. This showed how accurate our analysis was in terms of a successful project.

The LM317 voltage regulator was very efficient and more hands on and we were able used in converting the output voltages from 1.2V to 37V with 1.5A. All these values were derived from Texas Instruments data sheet. Other parts including the laser diode and the logic control of the system each had different power that may be needed.

This regulator provided helpful aspects to its room for more options to be chosen. We were expecting the IC to become hot during the operation and hence we decided in using a suitable heat sink to help in dissipating the heat. The circuit and the designed simulation for the regulator is shown. The resistors values to be chosen are as follows: Choosing our resistance value of R2=750Ω which was generally considered as the standard. The equation from Texas Instrumentation data sheet on how to use the LM317 chip is as follows. The resistor value derived below was used in the circuit analysis.

$$Vo = \mathbf{1.25}V * \mathbf{(1+R1/R2) + 50\mu}A * R\mathbf{1}$$

$$\mathbf{(Vout /1.25) - 50\mu}A\mathbf{= (1+R1/R2)}$$

$$R\mathbf{1} = Vout - \mathbf{1.25 = 1499.40\Omega}$$

5.4.6 Rotation Generation

The micro controller designed for this project used a PWM wave generator which had a limited voltage and current drive capability to control the motor, a motor driver was then implemented in order to meet the stepper motor's current and voltage requirements. In a scenario where the armature inductance of the motor was very large, the derived transfer function of the motor was out of a third order system, making the design of the system a complicated one. In order to avoid such a case from happening, the stepper motor was designed to a negligible armature inductance in order to derive a rotational control system of second order transfer function of the servomotor avoiding any complications of the project.

The main function of the chosen stepper motor was to be able to provide the amount of current needed for the torque to be able to rotate the chosen platform around its axis and have the ability in holding it in its final position for the power generated to be at maximum. From the motor's function we received a desired control signal that was a representation to the relative output position of the motor's shaft, and power was able to be supplied to its coils until the motor's shaft makes a turn to that relative position.

5.5 Software Design

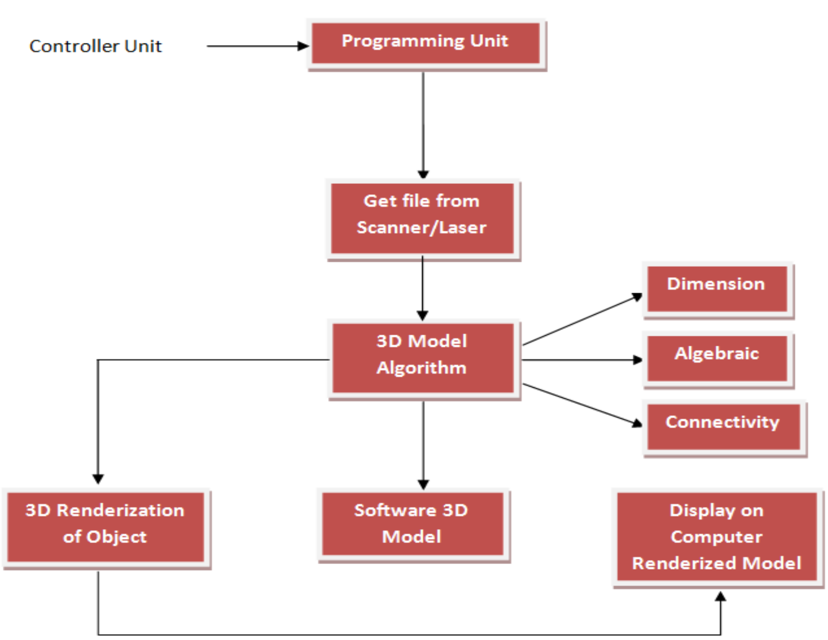


Fig 5.5 (1) Block Diagram of the Programming Unit

When we talked about the software design in our thesis we were dealing with basically every aspect of getting the data input file from the laser/camera to the data representation software. Also, we talked about a lot of aspects regarding previous works done in this matter of 3D scanning algorithms. In the following block diagram we will be discussing furthermore each and every part of the process of getting the image from the controller unit and passing it through the software that we will be using. Moreover, we looked forward to test many aspects of this important aspect of our thesis.

The idea was to basically get input file or point cloud from the controller unit and pass it Programming Unit, which was in charge of getting the images and process them computationally as well as making sure that the object was of a decent size, shape, dimension, etc. We did make plans in using Java as the programming language since there existed notable work on 3D Scanning Algorithm written in that language where we can take reference of to make our project simpler. However, our plan to add special features like adding color to the rendered objects displayed on software.

3D Model Algorithm builds upon earlier work in model synthesis. The user provides an example model ~~as~~ the input. The example model was a set of polygons that form closed polyhedral objects. Model synthesis generates a new model that resembled the scanned model gotten from the scanning process. Some factors in consideration or constraints that we had to deal with were applying the algorithm based on dimension of object since it was allowing the user to model the object in a realistic dimension. Another constraint was the algebraic part where the user will have to calculate algebraically the size, shape, and height. Ultimately, was the connectivity constraint where user will have to pinpoint the object where holes occur in modeling so that it does not exist incomplete points in the renderization.

In the 3D Software model we were planning to use Autodesk 123D. However, we switched to a more reliable or more robust 3D modeling software like MeshLab. Finally, after the scanned object has been synthesized and renderized, we used planning to display the smooth 3D modeled object in a computer screen via USB/FireWire Cable from Camera/Laser/Scanner to the CPU.

Furthermore, from our plan to expand on the actual design as we went along the semester. Specially, all this software requirements were tested and prototyped when the time came. From the initial rough sketch of how the program control and algorithms was looking like, we implemented further analysis

5.5.1 Camera System Communication

Camera System Communication refers to the method of communication between the obtained images from the actual webcam device to the controller unit and then to the data representation software passing through all the algorithms to properly identify the object we were going to modify. This was going to be achieved through a custom library that that was run in the Windows 7 operating system called Video Windows. This software was getting the image from the camera and laser, accelerated it and then the image was buffered and rendered for each subsequent scan.

5.5.2 Platform Communication

The whole communication software part was to be able to recreate a 3D spatial representation of an environment to use a laser/camera/sensor and a stepper motor as well as microcontroller units. We were able to obtain the data from the microcontrollers as point clouds to them work on the data representation software. We then used an ATMega328p microcontroller to communicate with every part of the whole 3D Scanner. This microcontroller was able to help us to obtain the necessary information and data files that we needed to actually model the desired object. The laser/camera/sensor provided the necessary communication and interface to the data representation software via a USB cable. We were able to program the microcontroller with simple commands to obey and then translate that to the software that we used to obtain accurate and appropriate responses.

We had plans to implement several functions like the rate to which we were going to be sampling the scans, other functions like controlling the angle of inclination, degree of field of vision and many more that we was actually developed later on.

5.5.3 3D Object Modeling

When talking about the 3D Object modeling part of this project we wanted to recreate or model the three dimensional aspect of the object as accurate as possible. That was why mathematically as well as geometrical aspects were considered in this part. The three dimensional space was represented as X, Y, and Z axis. In order to obtain the data from the microcontroller we first needed to inter communicate each device accordingly through a series of commands and algorithms. In this project we were able to utilize Atmega328p to receive commands in order to form the point cloud.

The information provided by these microcontrollers allowed us to be able to synthesize the horizontal and vertical depth portions of the desired object. Also, the data files were saved in memory and it was then erased until we turned off the system. That was why we were planning to retain this data files and hopefully save them in the computer so that no data or point clouds were lost for future use. We were also able to obtain a basic 2D sample which was also saved in local memory. The process of scanning was continued until we reached a whole 360 degrees around the object for perfect reconstruction.

Finally we wanted to minimize the time scanning as much as possible so that this process did not become tedious and increments performance of the 3D scanners. We were planning to scan the object in less than 1 minute. However, we still had to experiment with various cases. Depending on the resolution of the scanned object, more time had to be required to actually obtain the best possible rendered model of the object. Ultimately, more details arose as we went through the actual construction of the 3D scanner and its sequence of steps to completely get the best representation of the scanned object.

5.5.4 3D Renderization

Rendering of an object is just the process of generating the image of a modeled object. In our case we rendered the scanned object ideally to best represent the objects characteristics as well as real as possible from actual object. This included all geometrical shapes and dimensions of the real object in the data representation software that we were using. The final rendered modeled contained many characteristics such as different point of views by means of 360 degrees view, texture, shades, illumination, etc. Many programs were available to render scenes which was the term given to actually rendering, and it was used in video games and animated movies development.

Fig 5.5.4 (1) 3D Rendered Model of an Office
(Credit to docs.autodesk.com)

In our project we used data representation software called MeshLab which was free to use and easy to manipulate. There were also many techniques for rendering, however, we mainly focused on basic strategies since the most important part of this project was actually communicating the devices and obtaining the point clouds for further use. Later on, we actually implemented these so called rendering techniques to even better view the scanned object as real as possible digitally. Also, we made room for a needed high speed computer with a high quality video card in order to maximize performance when rendering the scene or the scanned modeled object. In the following picture we can demonstrate how real a scene was able to when rendered.

5.5.5 Algorithms

Our 3D scanner implemented several algorithms from the microcontrollers to the data representation software. Some of them will be discussed thoroughly in this section. We will begin by the microcontroller unit and then move on forward to the actual data representation software algorithm.

5.5.6 Microcontroller Algorithm.

The whole idea when designing this module was to actually control the power flow to the stepper motor, the camera calibration, the angle of inclination, etc. More design regarding this will be covered later on when we finish detailing with all the mathematical aspects of getting the image and all its characteristics dimensionally correct. For now, the next diagram represents vaguely how the final prototype was looking like. This is what we wanted to achieve and it was pretty straight forward, however, we had to adjust the design later when we had to for analytical observation.

At first, when we first planned to build this prototype, it consisted of a rotating platform fueled by a stepper motor that rotated 360 degrees while the camera and the laser captured and scanning the desired object. Some interesting facts were generated on here. First, we had to calculate the alpha angle to which the object was going to be scanned and captured which was perpendicular to the rotating platform. In this matter, we had to come up with a decent and not so robust algorithm to actually implement this first prototype which changed as we went along this thesis

This was the ideal design that we wanted to achieve using laser triangulation. The difficult part was when we were trying to put together all the parts of the 3D scanner, especially getting that motor to rotate 360 degrees around the object. A well designed rotating system was required as we went along this project and started building our prototype. For the processing unit part, the most important part of the software was in calibrating the camera since it was necessary to obtain every possible point of view of the object as well as the angles. Before calibrating we obviously needed to correct measurement of the position of the hardware so we can adjust the software accordingly.

Another algorithm that we implemented was the laser intensity. PWM pins of the Microcontroller was able to permit that, as every modern day microcontroller has voltage readers and analogue to digital converters and vice versa. In the documentation of each device we were able to find this information, more specifically in which memory addresses those related instructions were available and how to enable these extra functions of the MCUs. These were the base for developing algorithm, hence we will be going in detail in the working programmable prototype. We had some pseudo code; however, it was implemented later on when everything was put together.

5.5.7 3D Modeling Algorithm.

This was the most important aspect of the software design. First, to model the object geometrically and mathematically correctly we wanted to find the Cartesian coordinates in the three dimensional space x, y, and z that belonged to the object. Basically, we had to calculate for the distance between the rotating axis and the point marked by the laser/camera which was marked as 'r' in the following diagram. Also, we need to find 'θ θ' which was just the angle between the laser and the camera using simple calculus:

$$\sin(\theta) = \frac{r}{\phi}$$

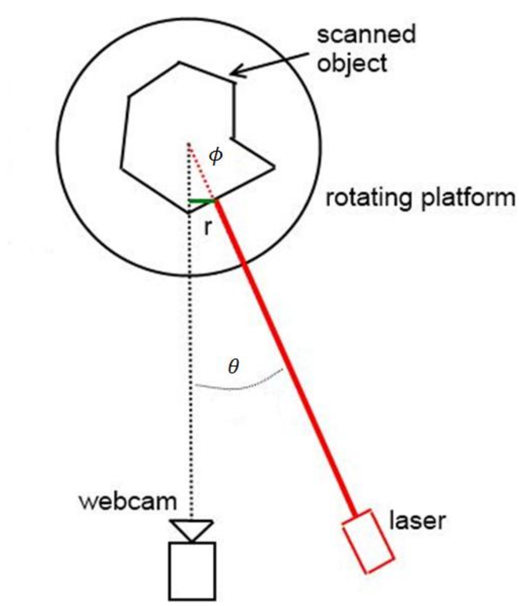


Fig 5.5.7 (1) Laser Triangulation Diagram-A

This operation was designed to be able to do it hundreds of times if not thousands to get every possible angle of the rotating object to better model it by iteration. Since this diagram represented only two dimensions the following picture will represent the actual three dimensional space which we worked on.

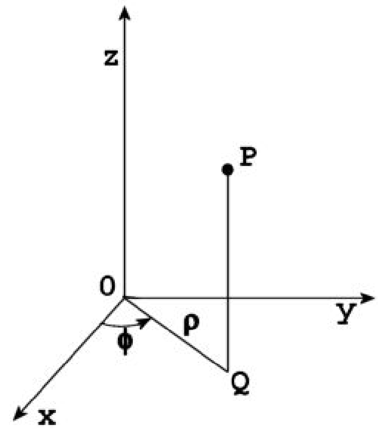


Fig 5.5.7 (2) Laser Triangulation Diagram-B

As object rotated in the platform we wanted to update 'p' for better calculation. For example, first iteration P = 3, second iteration P = 6, and so on. In order to get the best picture of the object and accurate scan of the object as possible. These were some ideas that we had been working on . However, some changes was made to occur as we went along. Finally, this was the principle that we entirely be applying throughout our entire thesis.

5.5.8 Preliminary Testing

We wanted to test this prototype as soon as possible to become familiar with all this new information that we had gathered along this time. We wanted to test this 3D scanner in a closed environment, not so much light, obscure, so that no interferences in the scanning happen. Also, we used a laser, a camera, a servo motor, two microcontrollers, a computer, and data representation software.

In the prototype phase, we tried to recreate the whole environment of the final device. Instead of circuit boards, we had everything built on a breadboard, except for the power supply. We then had a set up specifically designed to emulate the triangulation method, from which we were able to deduce all the physical distances and configuration. This was a critical step that we couldn’t bypass or put aside for later, as the development of the software was depending on it on every phase of it, from calibration to renderization.

5.5.9 Programming Languages

We used AVR assembly C to program the controller and modeling algorithm. For the processing unit, although Java does not offer the best efficiency for this project, it was the most reliable and easy to program. C++ could also have been considered, however, the code would have been very robust and not so efficient.

We ended up going with C++ for the final program, and the only reason for that is that the open source libraries for computer vision and image processing were written in that language.

5.5.10 IDE

We planned to utilize Eclipse as the programming environment because of its simplicity and several years of familiarity with it. It generally allowed us to create simple method which interacted with the overall system. Debugging the program was pretty straight forward since there was plenty of open source code which we made reference to.

Finally, Visual Studio 2012 for Windows and its equivalent for Linux like Qt 5.0 Creator were the ones chosen for our final program. A linux machine was our platform like Ubuntu, as C++ became our high-language of choice for this project.

5.5.11 Testability

Testability and maintenance played a huge role in this project, since the code was the major part of this project, and it was made easy to be read, and easy to modify without changing the whole software design. From previous knowledge, we were able to have the code written not to actually develop something but to be able to maintain it for the rest of the project because of major precautionary Measures associated.

With such an iterative software process, testing the components as we went along the project was required and common. Testing the iteration for specific scenarios was also required for this whole thesis. We wanted smooth and reliable coding to run on the microcontrollers in order to get the best performance and accuracy. This was why we wanted to use object oriented code as much as possible. It made it easier to develop our classes and access which part of the software we wanted without affecting the other parts too much.

Finally, we included a user manual in order for future users to maintain the 3D scanner by themselves. This project was really straight forward in terms of what it does, how it does and when it does it. We only had to put all pieces together to finish and assemble the prototype which was a success.

Trial and error was a big part of our project because we tested out the computer vision algorithms in-real time while trying to make out project work. The programming interface was always open while testing and we constantly changed the code as we were running scan after scan.

5.6 Construction

The integration process for our system depended heavily on programming the microcontroller to interact and communicate with each component. The code for our Arduino was written in C language and utilized Arduino libraries. The code started by defining the necessary constants that were used later in the process. Then it initializes the serial communication with a baud rate and went on to initialize the servo motor controller onto a digital pin. The main loop of the code was made up of for loops that went to determine motor position from 0 to 360 degree. During the sweeping process, at each increment of the loop, the camera was used to collect data and stored inside of an array. Once the loop was completed, the processor then checked if it's covered the 360 degree. The data was then formatted and sent to the serial channel to computer via USB.

At this point, the data package arrived to the computer's COM port and had to be parsed and read. This was done through by MESHLAB. That was checking the buffer for characters. Once the buffer was found to contain data, the script splits the data into separate values and stored the camera readings into a new array and stored the number of scans completed. Depending on what the number we got, we sent signal to motor control module to rotate the platform slightly and repeated the process. For every turn we took two images, which were taken from the same angle. In middle of every scan we kept track of what angle the object was pointing and quality of the previous image. When the image was not good, we rotated the motor one step backward to retake the picture. All this information was needed to be sent back to the computer with images. We needed to recreate the image.

Now that the data was stored, it needed to be converted into the appropriate format. Once you have two images first image was taken with laser on and second was image was with laser off. The raw image reading was subtracted from a second array. And created new data that was only where the laser was pointing on the object. All the images were different lines at different position depending on the angle the object was being pointed at. In one image, we were able to break it down even more in the horizontal slide and created ten to twelve new steps. Then an algorithm was then used to calculate the distance for each point in new image which was just vertical red dot. By breaking down more step resolution went up but the computational time also went up. Once the all of the values are converted, in the entire two hundred step, we keep track of degree the image was taken from because we needed to rotate the point manually and then used to plot all of the individual points in the three dimensional environment. In fact we simply sent this data back straight from the MCU unit to the Arduino. We were simply incrementing this number inside the memory for which we were able to find out the address and have access to whenever we needed to. The way we set up our design was with a two way communication between processing and motor control units, so we justified our advantage of an already set up system to aid us in software development. That value, the number of steps, and consequently degrees, was then accessed in Arduino whenever needed to process our images, and that eased the PC renderizing part, since we were able to obtain raw files with more crucial information for our needs.

After all of the points were plotted for the set of data and created mesh cloud, the mesh object were created (small triangle using three points) and then added to the data structure, which gave g rise to 3D object like quality, this happened in other open source program. Once the scanning was over, the final mesh was then redrawn to finally show the final model which was then saved and exported in some formats. This allowed for the object model to be used in other applications. Since we were using multiple different software the entire step did not happen automatically, we needed to do some part manually. Also during creating mesh like structures, it was not automatic and when it came from fixing holes and editing the point mesh structure, we then did it manually.

5.7 Testing

A major part of the testing was done in a prototyping phase. A large breadboard was used to recreate the final product that we designed, notably what was inside the PCB. We had all the inputs including the power supply and its ground on two different conducting lines, this enabled us to have an easy access to every pin we needed in our 3 integrated circuit chips, the MCU, the serial communication converter chip and the motor driver. Plus, It made it easier for us to put the final touches on our hardware design, to figure out which was pulling up resistors we needed and various capacitors to minimize noise and other problems, optimizing the product. By running the prototype with the desired emulating set up, we tested out our devices in different ways, from the hardware to the software side.
As soon as we received the PCB back the fabrication house, a digital multi meter was used to verify every single connection with the traces according to out schematic. There were no problems of this side with the PCB. The same was done again after we populated the board with both thru-hole and surface mount components. Re-soldering was sometimes necessary, but in genral there was no major problems on that point either.

Due to our design being modular and we had some prepackaged electronics; testing could be done on the individual components and verified. With systematically testing and verifying each component worked we were able to focus on components that were not working properly. Some generic problem where it had nothing to do with our component could still cause problems, like in one of our components that was the Atmega328p, where sometimes some open source library interfered with other libraries where they may be used to do the same variable name or reference function. A Library with poor documentation could had made it harder to find problems, and also the data that's coming in under a different value than what was specified and couldn't be handed by chip could have caused the issue, or the stepper motor could randomly flair inaccurately and the serial input data wouldn't make sensitivity to test the system under different conditions.

It would have shown us constraints of the system. We were able to measure the response time of the system. We were using multiple different software, before final output file was finished and doing some step manually it was hard to figure out the response time of the scanner. Due to heavy use of the software, testing incurred to test all race conditions in the system. We tested the reliability of the software with various inputs.
We also considered different environment and light condition to do testing in.

5.7.1 Primary Control

As the brain of our 3D scanner, the function of the MCU was sending control data to other modules to trigger actions depending on current state and inputs. These modules were the following:
1) Send handshake back the programming unit
2) Send control data to the laser
3) Send control data to motor system for motor to turn a certain degree.

The procedures were to analyze digital logic, we needed a logic analyzer which was connected to the controller output. And used the function generator as the input. Then tested various inputs to verify the correctness for each possible input combination. For the second case we had to check that the data line was high while idle, and then sending a known input from the function generator to check if the corresponding output was what it should be.

5.7.2 Processing

1) Convert deviations in laser position into relative distance with 0.3- 0.5cm resolution. 2) Store and relay distance points in a computer accessible format. 3), perform calculation was done under reasonable time.

Procedures - by loading an ideal input (black and white image with red line in middle), in the processing module. Which was MATLAB; we then calculated with program and compared the output distance with the measured distance. For Second case we had to load file that we knew it was correct and fed it into our program and to verify if output came out to be as it should have been. For the third point, we used the same method with ideal files formatted with known output and run it into our program and measured how long it took.

5.7.3 Motor Control

1) Motor driver rotated the object consistently by 1 to 3 degrees per step. 2) Motor was made to have enough torque to rotate the platform with object on top.

Procedures - by placing different weights on the platform and marking a reference point on each object. And connecting the function generator to the motor driver and run the motor for fix number of steps, then measured the angle of rotation between the start and marked reference position. From the initial startup, the stepper motors did not have to make a movement to the home point or position. This was due to the fact that when the whole set up was initially off, the motors were able to move to any arbitrary location. The micro controller didn’t have to know the exact position at that moment. Yet extra verification was needed during our prototyping phase. This was done with different weight objects to test reliability.

With the micro controller being off, the machine was initially turned on and the micro controller was be able to generate a movement for the motor to make a backward movement until it hit the home position. As soon as the motor made a specified direction to the home position, the stepper motor finally stopped and was able to receive various input commands from the computer. From this step, the stepper motor was able to detect when it had reached the home position. This verification was made possible due to the fact that it had a limit switch was associated at the end of the rails. This generated the detections when the motors reached a home position.

With the axis making a maximum limit switch, the switch was able to generate a signal to the micro controller which in this case was able to provide a low send and the controller was be able to make the stepper motor stop. This circuit being designed for this limit switches was able to contain a low signal noise. This was due to the fact that the signal noise level generated was a bit too high which raised a false limit to hit and then made a zeroing of the axis which and this was not useful. This approach was useful in the sense that it was able to be used to prevent the device form high levels of noise being made into this system and connection was made with a pull up resistor to the pin that was designed to be used for the input.

A capacitor was also added to the circuit design with the available resistors in order to introduce a reduction in the noise levels. We planned on using a 0.1 microfarads multilayer capacitor. This was be able to provide a limitation to the noise that came out of the stepper motor and the other set up. A limit switch was designed and this was mounted on the relative locations of the axis. This was due to the fact when input command were made to our micro controller, this was made to be incorrect or a false signal being generated as a home position. This in turn made the stepper motors not breaking down the set up. The micro controller was made to be able to provide data to the computer when this error occurred. The computer was designed to be able to send ability to our micro controller the direction to the home position. The code was rerun and the same code was made to run when the micro controller was being turned on. The micro controller would be able to keep a maintaining mechanism on the relative positions of the axis. This was of importance when the computer would have to send a different position to the controls. This could be an absolute position or a relative position.

The microcontroller was able to make the stepper motor generate a full step movement as well as the micro steps. This stepping space of the micro steps was made to be around two micro steps in a major step. This value was derived by making each major step for the stepper motor to be around 0.9 degrees and the gear was driven on the belt to a quarter of an inch so for each major step being made,. The entire movement was made to be close to two thousandths of an inch. This value was rounded to the nearest decimal place to allow flexibility to be made in calculating pulses widths to make micro steps. Even though there might be a concern for accuracy from the motors. Careful approach was made in debugging the errors. That way, we achieved a stable system to work with when prototyping. These changes were made necessary when needed and was given by the software side, as we decided then that that angle between snaps was not enough making it up for a less precise rendering of our final point cloud.

5.7.4 Laser and Camera

1) Single line laser created vertical line 1 foot away and 25 cm in length. Vertical laser line was clearly visible over the 1 foot. And also visible on object and covered the whole object vertically. 2) Image from camera was made to be able to capture line laser reflecting of the object being scan. 3) Whole captured image was made to be reasonable size that could transfer to computer easily by processing unit.

Procedures - We just used a ruler and measured the length of the line with fixed distance. And then for the second time with object being 1 foot away from the laser, and checked that it was clearly visible and repeated with varying distance. For second case we repeated the above test on various objects and surfaces to verify the quality of the image we took. Then we went ahead to measure time for long it was going to take image to transfer to computer from the camera.

In determining the angle at which the laser was to leave the line-generating lens, we calculated it based on the 1 foot prediction (30.48cm) and the 25cm line we needed. Using the inverse sin function, we came up with a 48.4 degree output angle, and that angle was then reduced if we took more than 1foot distance laser-object, and augmented when we shortened that distance. After we had acceptable laser-camera measures and setups, we adjusted the processing software settings as much as we desired to have optimal results.

5.7.5 Power Module

Power supply provided appropriate power levels to all specified components of the system.
1) 3 to 4 and 2.5-3.5mA for the laser,
2)10-30v and 350mA-1A for the motor,
3) 3-5.5V and 0.005-0.020mA for the motor logic driver.

Procedures - By measuring the open circuit voltage and closing the circuit, current was verified from the power module. By connecting voltmeter to the laser power output, and then verifying that the voltage displayed on the voltmeter was falling under right range. Then we connected a small resistor across the laser power output, Connect the voltmeter across the resistor. And calculate the circuit current using ohms law. We went ahead to use same method for motor where we connected the leads of voltmeter to the motor module. Verified that the voltage displayed on the voltmeter was in the right range. And then placed a register 0.05ohm resistor across motor power module. And connected the voltmeter across the resistor to measure the voltage drop on the resistor. Using ohms law, verification was made that the computed current fell in the range. Then we went ahead to do same thing for motor logic driver and verified the voltage and current.

6.0 Executive Design Summary

Our 3D scanner was designed to make a 3D model of everyday small object with some precision. More accuracy was better (width, height, length). It increased the effectiveness working with complex parts. Also it was able to be used for reverse engineering with precise digital model of the objects to be reproduced. Collected 3D data was useful for a wide variety of applications such as movies and video game. Also, it was able to combine use of 3D scanning and 3D printing technology which allowed the duplicate of physical objects. Since the fields of application are many, there were different types of 3D scanners based on different technologies but most of them were too expensive. Our goal was to make an inexpensive and reliable 3D scanner for small objects.

There were many different approaches to 3D scanning, what we wanted was to build an ideal short-range scanning (<1 meter) also known as Short Range Laser Triangulation 3D scanners. It used a laser line to scan across an object. A sensor picks up the laser light that was reflected off the object, and using trigonometric triangulation, a computer (program) calculates the distance from the object to the scanner. By knowing the angle which lesser point to object and angle between camera and laser, we were be able to calculate the distance from the laser source to the object's surface. By fixing the laser and camera to fix position. To get the 3D view of the object we have to rotate the object on its y-axis.

6.1 Hardware Design

6.1.1 List of Specifications and Requirements

- **Controller Segment:**

Various functions existed for the controller including sending informational data to computer for processing and data to the motor in order for the circular platform to rotate about its center.

For appropriate input combination an analyzer was introduced and connected to the controller as an output. For accuracy, more inputs were introduced to check the correct match on the logic analyzer and consistent high data check was done using the analyzer.

For the controller part we were planning to use the ATmega328 instead of the originally planned ATtiny45 and then we connect it to the programming unit via the FTDI FT232RL communication chip.

- **Motor Segment:**

Obviously the motor needed enough power to turn it on and have the ability to rotate the object of interest.350mA-1A of current and ±12 V of voltage was an ideal requirement.

A voltmeter was used to find the output voltage from the motor. This was achieved by placing the voltmeter to the motor leads. The current was measured from the ammeter in series with the motor

A critical stage was connecting the function generator to the motor driver part. Doing this, determined angle of rotations from the stage (object to be placed on for rotation). Noted spots were made on the stage with varying objects. The angle of rotations was achieved after running the function generator with respect to the noted spots, before applying this on the software side by programming the MCU controlling the driver. We sent pulses with the function generator with voltages and frequencies that was similar to the outputs on each pin used of the microcontroller. To get an appropriate current value to input while testing, we used a resistor to reach the 10 mA value as specified by the Atmel documentation.

We had decided to test several stepper motors like the ones discussed before, we selected the most appropriate and depended solely on the performance as well as the easiness to program and work with and not so much about the price.
We chose the one we originally planned because it worked with a very high range of voltages, which gave us flexibility power wise, and was good enough to hold as heavy of an object as we wanted.

- **Laser Segment:**

With varying length, the laser beam was scattered onto a vertical line. Approximate single layer beam spread could be from 30cm to 80cm.

Visible line was highly achieved over the 80cm line. Reflected LASER signals were collected and sent through a fire wire card to the computer and by means of Sensor Image Outputs.

For our laser we had spent a lot of time researching and came to the conclusion to use the best one available in the market and for its price was going to reduce the financial part of this project as well as maintaining performance, quality, response time, and was working perfectly with laser triangulation which was the base and principle of this project. The most difficult part was determining how to actually put together everything specially the laser part since it was the most sensitive part of the hardware. Also, we spent a lot of time making sure that the laser scanned completely the object to be modeled in the software part.

The laser was made by using short range and it was going to be non-contact based since we were planning to scan objects about 1 foot distance from laser/camera to potential object.

- **Power Segment:**

The power part of this project required a lot of skills since we were planning to design the PCB to provide power to all the components with the correct amount of voltage and amperes so that they would work within the boundaries specified below, and of course that they don't burn since the parts were going to cost a lot of money. We couldn't afford to burn our parts since we were paying for them.

Optimum power supply was to be achieved to supply enough power to all the parts.

- 3-4V and 2.5-3.3mA for LASER
- 2.3-3V and 7-9mA for camera analog power
- 1.5-2.0V and 9-12mA for camera digital core
- 10-30V and 300-400mA for motor driver supply

To achieve sufficient voltage and current supplies, the voltmeter was connected to the power segment and adjusted till the required measurements were achieved. This step was done for open circuit analysis.

For the closed loop analysis, a resistor was connected across the laser power segment and using a voltmeter, the voltage was determined. From Ohms law, the closed current was achieved.

Using the same analysis, the open circuit voltage and closed circuit current was able to be achieved when the voltmeter was connected across the camera segment. Same steps were used for motor supply power and the motor driver logic power.

The hardware design parts designed, took us a lot of time since it made in such a way that it could be varied depending on the difficulties that we encountered in the next part of the project which was when we started prototyping all the information gathered throughout this thesis. First we had to order the parts and started playing with them. Making decisions on which the various parts which we picked depended on how difficult the assembled parts in order to make a nice looking final product. Like we said before our plan was to not occupy a lot of space for this 3D scanner. Since it was short range scanner, most of the objects that we were able to scan in testing were going to be really not that big. Our plans to assemble the final product in Senior Design II was totally completed

6.2 Software Design

6.2.1 List of Requirements and Specifications

First, we have to start by saying that this was the part that took the most time during the developing this project since it was the one that had the most impact for the final result of the scanned object and further representation of it in the data representation software. Therefore, our plan is to give a brief summary of what this part looked like in the following part below.

We planned and used the most important part and principle of this project which was the processing part and it was to be implemented using the laser triangulation algorithm. Implementing this algorithm was a bit difficult since we had no previous knowledge in this matter. This was the base of the project in terms of programming.

.

We already discussed some mathematical background regarding this part of the programming part of the project; therefore, we will not discuss it here. Once we implemented this algorithm, we were pretty much done for the most important part when talking about the programming part of this project.

There was also a lot of programming in the controller part since it was designed to be in charge of controlling the servo motor especially as well as the laser and camera communication. We spent a lot of time doing this part also. Specifically, we needed to implement several classes and module to control each part of the whole system.

Some parts that we were able to program were the calibration of the motor, calibrating the camera, the rotation angle of the laser, communicating the camera with the microcontroller, moving the servo motor, turning on and off the system, controlling the power supply as well as making sure that this gave enough power to all the other components of the project.

We were able to scan the object, modeled it, and represented it in the data representation software in less than 10 minutes, giving us a real and smooth digital object as well as improved performance and response time. Also, between 190 and 200 times when the laser scanned the object in terms of loops.

In terms of what programming language, we used C++, since it was the most comfortable with and we had previous experience, also we selected this languages because there was a lot of previous works done in this matter, so that we made a lot of reference from. This language was selected also because there were a lot of libraries regarding this project that we used and became comfortable with, this allowed us to start playing with them. C and Java were also considered. However, C++ offered the best performance despite of being more robust and difficult to program.

We used an IDE Visual Studio 2012 for Windows and Qt Creator for Ubuntu as our working programming environments. For which we applied some basic pseudo code that was able to get our project beyond our expectations. It was just a matter of starting coding and putting every piece together to achieve a fully operational short range 3D scanner.

6.3 System Building

For our process to be optimal, we had to account for a few factors, but one of the most important ones were physical stability. For an accurate scanning, we needed to build our 3D scanner in a solid manner so that our results would be reliable. That meant that the moving parts had to be moving in a constant, predictable manner, and the nonmoving parts to be fixed so that they were able to do their job the way we designed them to be.

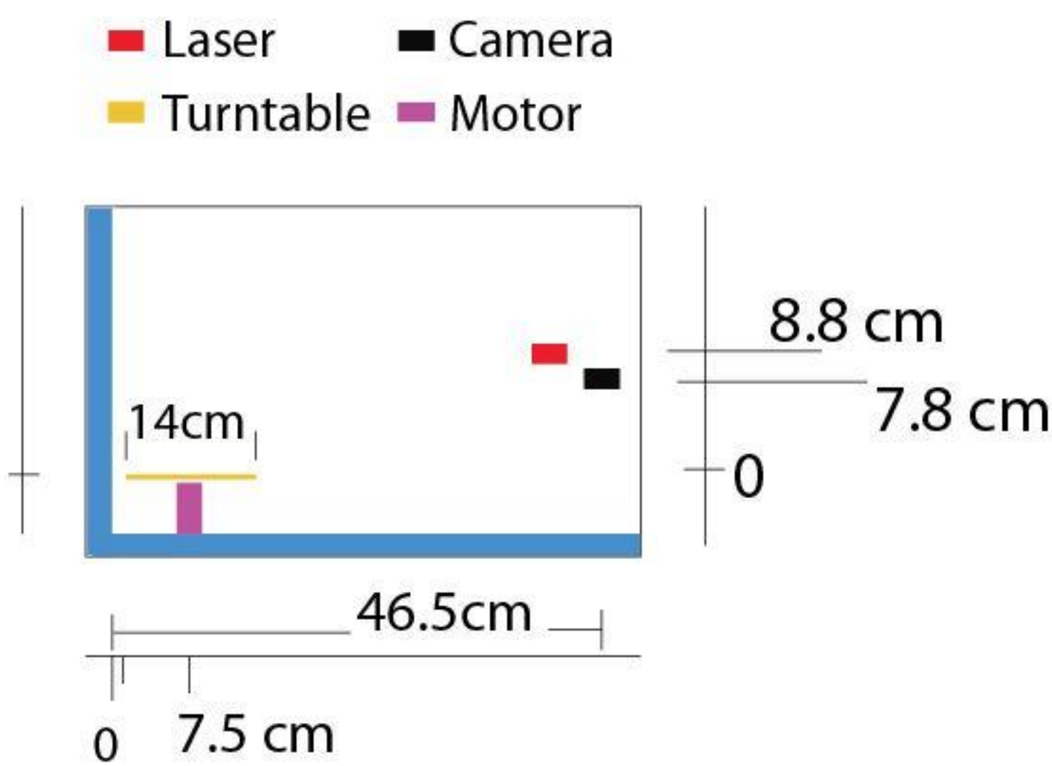


Fig 6.3. (1) Side View of SR3D

We built our final platform to be a 19 cm x 47 cm. For the process to work we needed for the inside to be dark when needed while the scanning went on. This allowed for a better reading of the images, as each of them was composed by the laser illuminated surface and the rest as black. For this to happen, 3 sides of the cube was covered and the third had a door with an opening to visually monitor the process. It was still made to be dark in there and we had an outside look of the process.

Testing was an important phase for our project, because this was the only way to figure out where our components were and physically placed. As mentioned before we tested the functioning of the process depending on every element's position followed by calculations. When a position seemed right, we tested out the whole process from start to finish and checked our results. We repositioned laser and camera right after processing with the Arduino, or we could have made it to review our physical setting after the 3d model work on the PC. No matter what stage we needed to do so, fixing the object temporarily, with duct tape or paste was made (incase just carefully putting it down was not enough).

6.3.1 Laser

The laser module consisted of the laser device itself, which pointed towards the wall, going through the scanned object. Of course we needed that line to be vertical, which was why we included a half cylindrical lens at the tip of the laser. The beam hits in a perpendicular way the round side of the half cylinder, and went off the flat side as a vertical, thin beam, like a sheet in the air.

To isolate and fix the half cylindrical lens, we placed it between two wooden beams that was made parallel and fixed on the base, on which the laser went on it too. We were able to fix the sides of the lens to the beams for perfect perpendicularity of the plane surface and the wooden base. Once we were satisfied with the way it was set up, super glue, because of its transparency and practicality was applied between the sides and the wooden beams. It was the better choice, because nailing it would have damaged the lens or alters the optical properties required for a perfect, straight reflective flattening of the laser beam. Once the beams and the lens were set up on the base, it was time to implement the laser.

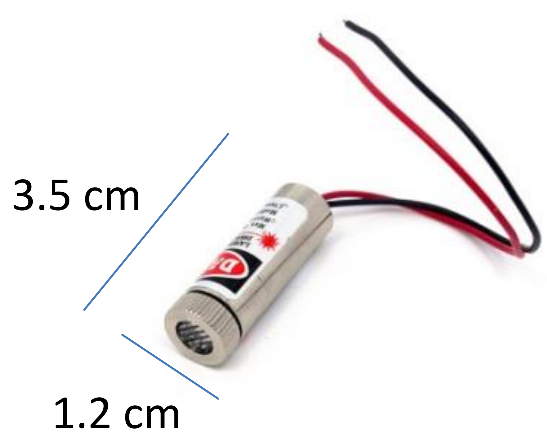


Fig 6.3.1 (1) LinienLaser
Approved by Apinex.com

The first step was to be able to position the laser in order to propagate the beam at the exact middle of the half lens, or at height H/2. A rubber base that we easily cut off until we reached the desired height seemed like an easy solution. Once attained, we needed to fix the laser pointer on top of the rubber on top of the wooden base. A piece of Duct tape over everything was able to do the job just fine, without making it impossible to change the positions once the testing phase started.

The camera was at the other angle of the triangle after the laser module. It was to be oriented to face the Z axis of the turntable / object scanned, towards where the laser line hits the object, in a certain way to always have the most visibility of line that was forming our point cloud.

What was finally done in our project is to choose a laser with an already built-in lens that outputs a vertical line.

6.3.2 Motor and Turntable

After we have our motor system finally figured out, with the PCB including control, driver, as well as power input all in a working state, it's time to mount it up into the motorized turntable. The only thing spinning here is the shaft of the motor. It will then be the base of the physical turning platform. The motor and turntable are going to be the one in charge of making to move the object to properly be scanned and modeled in the data representation software. For that we are going to select the

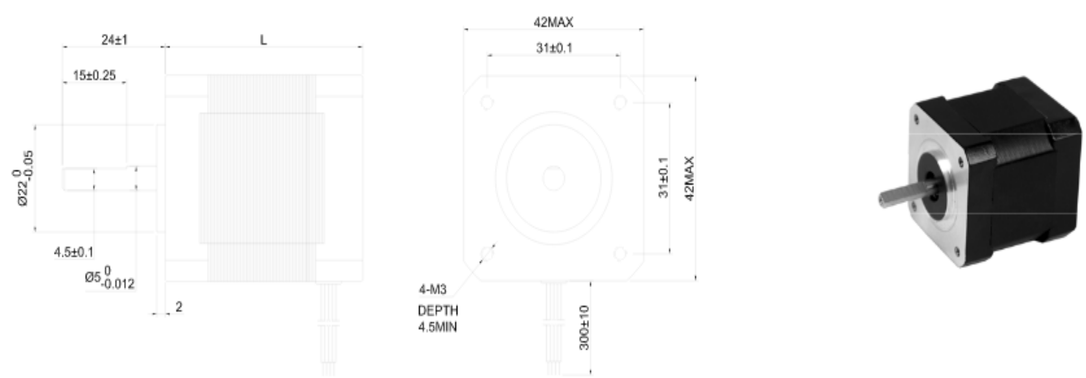


Fig 6.3.2 (1) Datasheet extract of physical specs for the NEMA17 motor (Credit to OSM stepper motors)

As we can see from this diagram, the shaft itself 4.5 millimeters wide, or 4.6mm including the maximum denture measure possible. This will be important for the design of the next part. We need a solid, easily worked with material for the turntable. Drilling a 4.5 mm hole in the center of a wood-like material disk would be optimal. Now while we can choose pure thick wood for this, we want to lighten up the turntable itself so that we can maximize the input weight of the scanned object that will go on it. This is why getting a plywood disk of 14 cm of diameter was optimal, as it is solid and will not be as heavy as wood. Also, to make sure the shaft hold the torque and doesn't slip as the hole become looser, we will clamp the shaft so that it will be extended by two thin metal arms, one on each side. We will then need, on top of the hole, cut a hemispherical vent, with a small saw, of the size of the clamped arms. This will allow to a better control of the wooden disk. After that, we want to maximize the grip on top of the turntable, to make any object stay the most stable way possible, immobile, while spinning.

Otherwise, the integrity of the scanning will be compromised if the object changes location throughout the scan, even a little bit. This is why we need to cover our material by a very grippy surface, just as a silicone or any form of thin rubber that we can easily cut into the disk shape and integrate to the surface of our turntable.

6.3.3 Components and Electrical Equipment

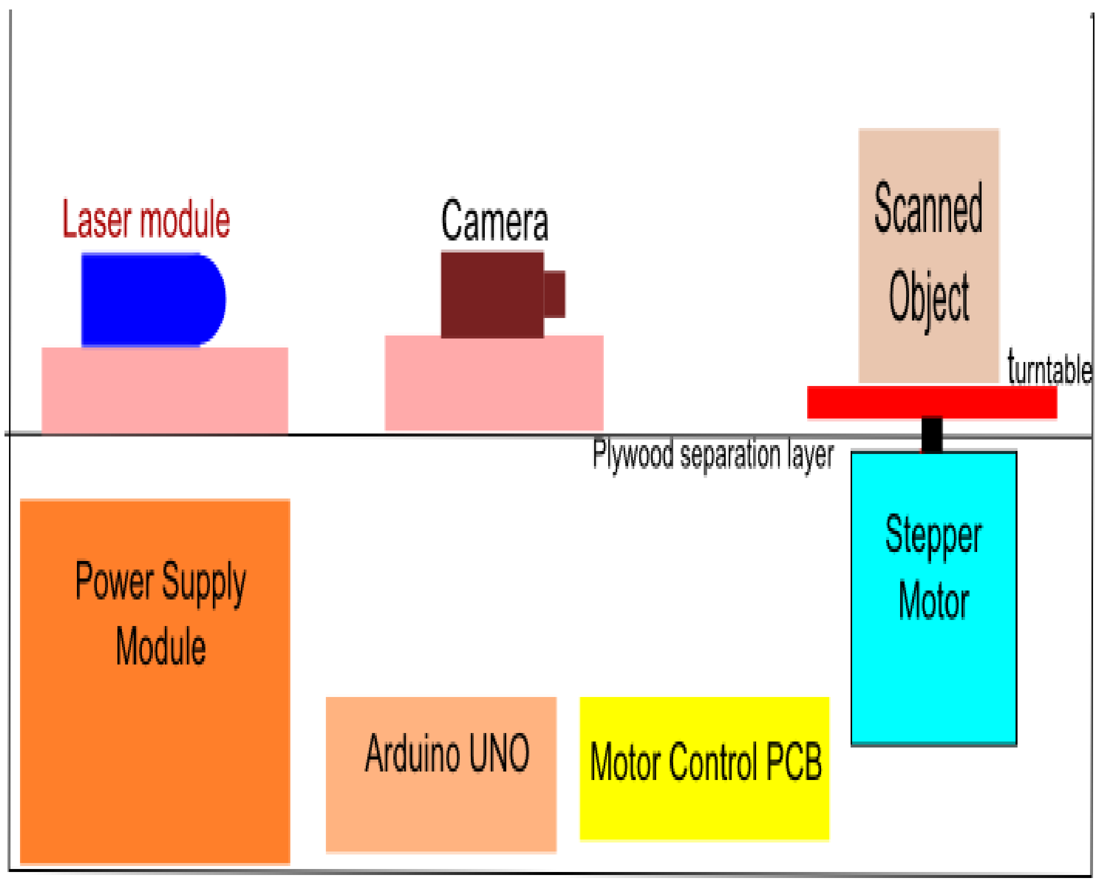


Fig 6.3.3 (1) Vertical cut view of the components diagram

Our components are:

- Printed circuit board containing the motor controller, driver, and communication chip.
- The power supply for the different devices.

Just as mentioned a few paragraphs above, the “brain” and power elements will be in a lower level in the whole frame, when the laser, camera and turntable will be on the higher level, or the 2nd floor if view in a building model. As far as connection goes, we will have the power supply unit with two cables connected to the Printed circuit board with 3 cables (Ground, A 5V input that will power the microcontroller and the motor driver chip, and a 9V input that will be driven to the stepper motor). 2 cables will connect the PCB and the Arduino, allowing for the transmission and reception of information between the processing unit and the motor control unit. Holes on the separating sheet will be drilled to allow the powering from the supply to the laser as well as to the camera. We will reserve another hole on the closest wall to the power supply, to let in the wall plug cable powering it all.

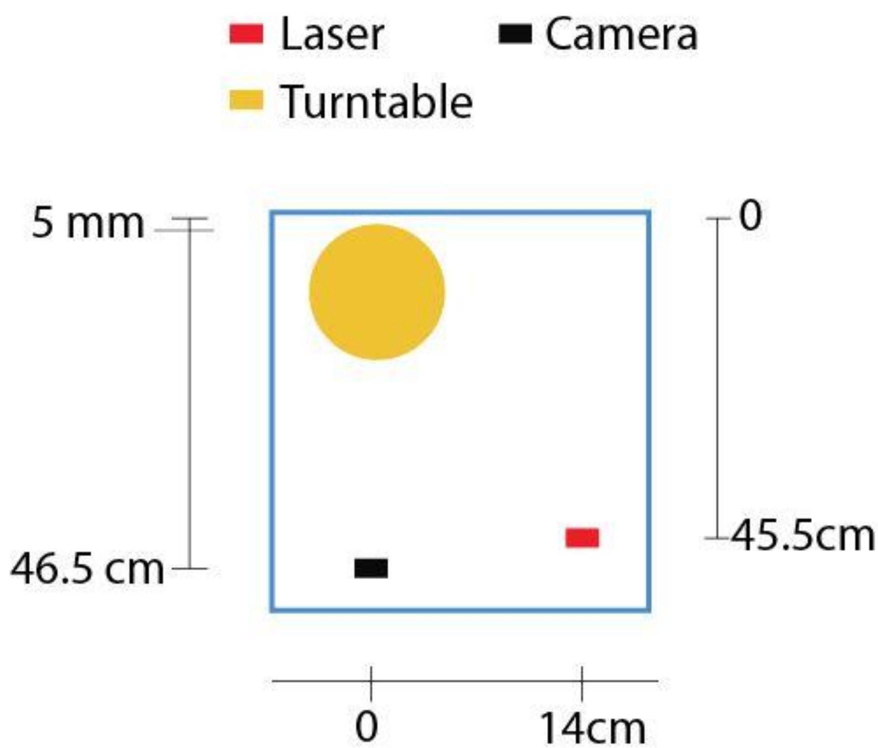


Fig 6.3.3 (2) Top view of SR3D

We will need rectangular platforms to hold the components in place, as well as the motor itself. It will come with holes of each corner so we can screw it the desired platform. It will be designed so that it sit just below the separation sheet, so that its upper part, the shaft holding the turntable disk, goes through the plywood that we will drill at that point, to make the motor hidden from the top and the turntable operating above the plywood sheet.

Just as described in the triangulation concept explanation, the top part of our scanner will be composed of the laser module and the camera, both facing the object on the turntable. Behind the turntable will be our back wall, which will consist of a white sheet on the adjacent wall to it.

6.3.4 Programming Unit Class Diagram

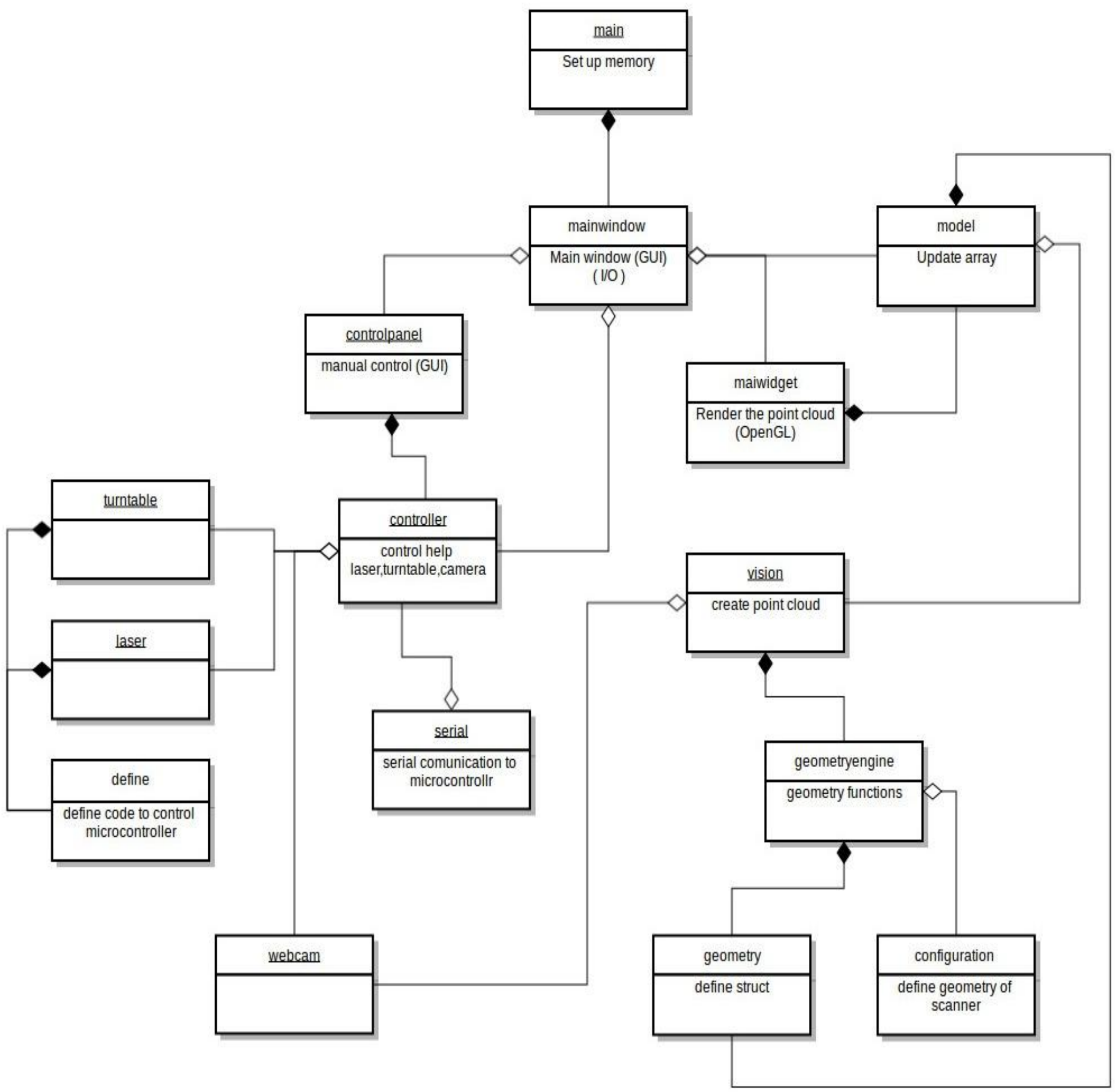


Fig 6.3.4 (1) Class Diagram

6.3.5 PCB Circuit Layout

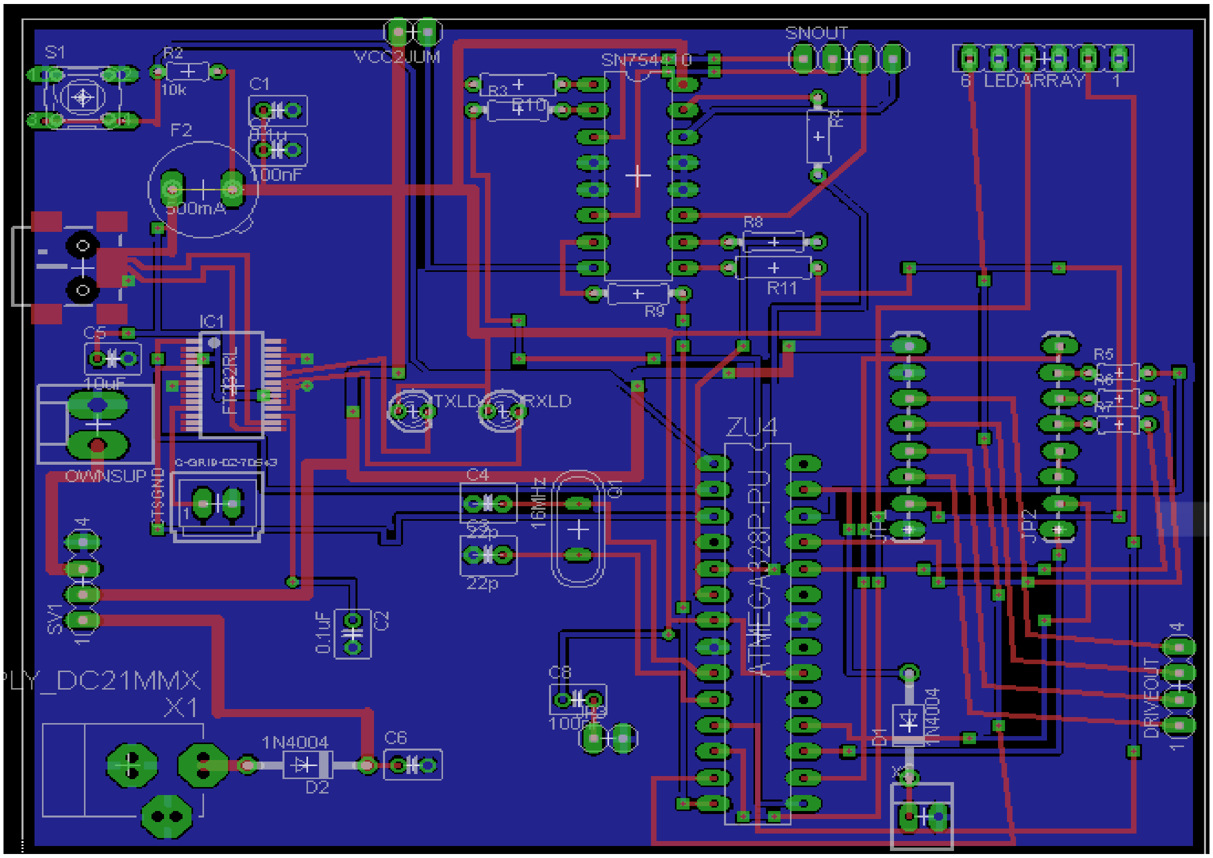


Fig 6.3.5 (1) PCB Circuit Layout

6.3.6 Project Operation

SR3D is comprised of a hardware and software part. The main components are: a camera, a line laser, a stepper motor, a platform, and the PCB above which controls every component. However, the most important part is the software, which is the one that is going to represent the object digitally by adjusting the features by the user's needs.

For example, in the following picture we have the main control panel of our software which allow us to adjust manually the field of view of our physical object, the angle of capture, rotating the platform by certain degree, detecting the laser, turning it on and off, moving it, etc. Also, it allows to fetch the main image, or starting image, to which the scanner will start. It has to be an empty image, so that no outliers are present. This is manual setup of the scanner. Now, the automatic setup will just consist of taking a reference image, placing the object in the rotating platform, and just press the START SCAN button. After around 8 minutes, a fully digitally sized object will be displayed in the form of point cloud.

Fig 6.3.6 (1) Control Panel SR3D

After the object has been scanned we proceed to clean and manage the point cloud in MeshLab. We use this software to eliminate some outliers that the scanner might capture.

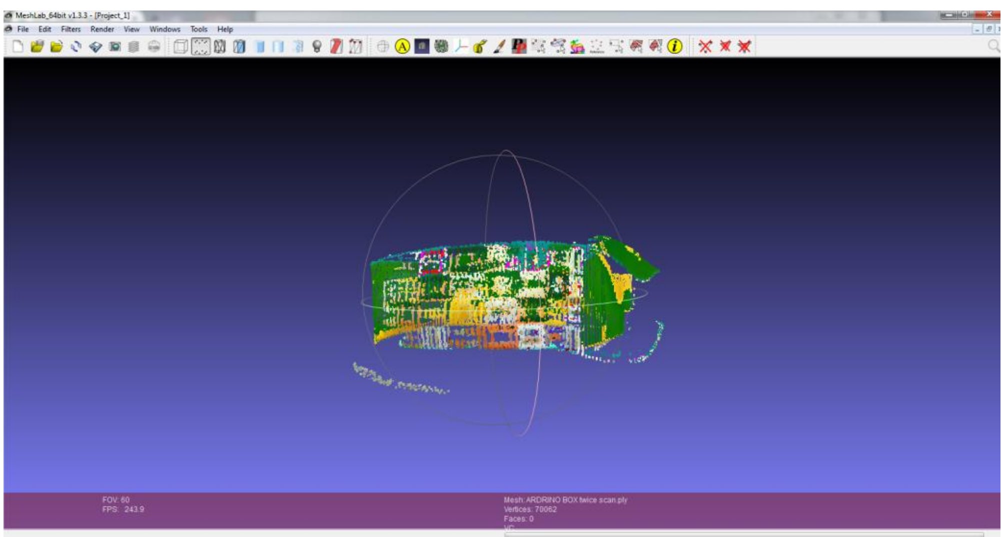


Fig 6.3.6 (2) Point Cloud Post-Processing using MeshLab

6.3.7 Testing

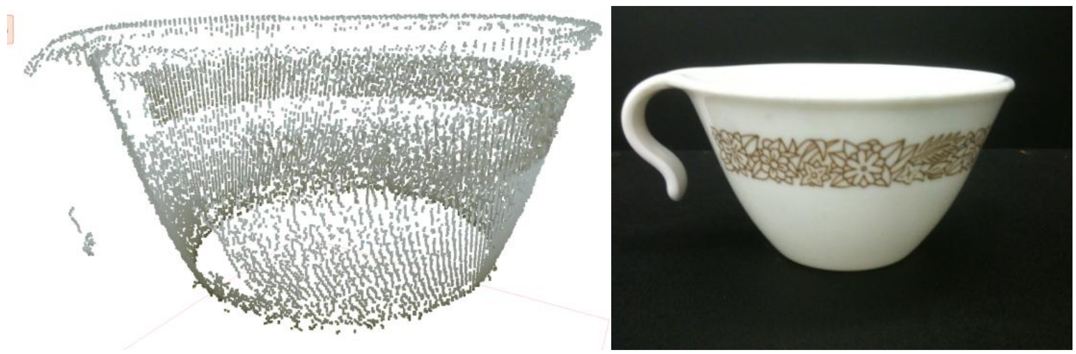


Fig 6.3.7 (1) Testing

Fig 6.3.7 (2) Testing

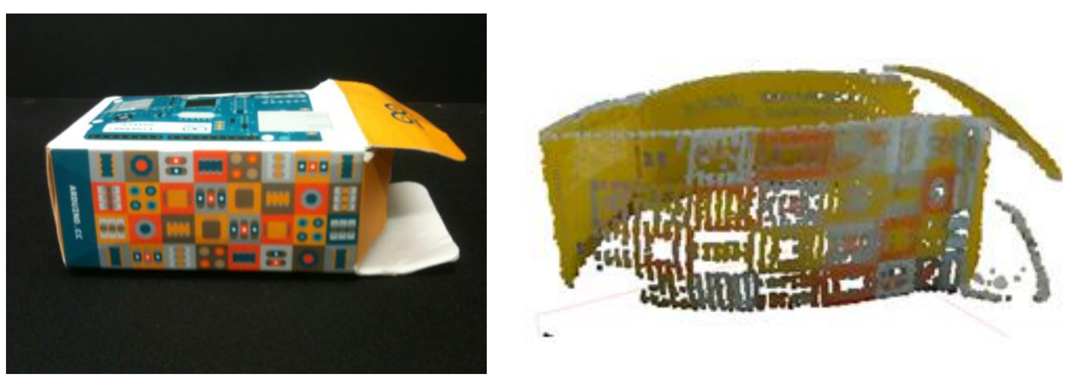


Fig 6.3.7 (3) Testing

7.0 Administrative Content

7.1 Budget

First, the base of our budget and finances discussion starts with the initial amount of money we want to put into this project. Based on the regular book budget for textbook in the two semesters of the senior design class, we get about $100 each, totaling $400 for all four of the members. This is an arbitrary amount, and only after our cost analysis we can talk about allocating a higher or lower amount. This analysis will be mainly focused on the parts we will need for this project. Also, we will be paying out of pocket since in this term we could not find any sponsor.

7.2 Financing

We are planning to spend less than $400 since most 3D Scanning Devices in the market are around that price and we will not get sponsorship for this project; therefore we will have to obtain the parts and buy them by ourselves. The money we will come up with is expected to be from the personals funds of the four members of the group as discussed earlier, like if it was regular textbook money to dedicate to college classes. Although, an option not to ignore completely is to be sponsored by a company that could finance our project in exchange for improved university relations or extensive publicity, that could attract potential clients and qualified employees. This is however an option not to rely on for now, but to definitely keep in mind as our project goes on.

Item	Part #	Vendor	Price(estimate)	Price(so far)
Arduino UNO		Arduino.cc	25.74	29.99
Motor Driver	SN754410	Mouser Electronics	2.99	2.99
PCB fabrication		4PCB	33.00	33.00
Solder	AT31504	Amazon	10.00	10.00
Stepper Motor	17HS15-0404S	Pololu	19.95 x 2	39.90
Line Laser	01012561	Miniinthebox.com	7.99	7.99
Web Cam	C270	Logitech	21.99	21.99
Power Transformer	DL-36-2	Skycraft	34.99	34.99
Voltage Regulator	LM7805	Adafruit	1.99 x 3	5.97
Box(parts)		Home Depot	50.00	74.99
Electronics (parts + Tax + shipping)		Various	68.87	68.87
Total			<400	$330.68

Table 7.2 (1) Financing

So we come to a total of $330.68. This is about $82.67 per member of the group, we did not have to spend more than $400 which was our initial thought.

7.2.1 Parts

First, our main mechanical component would be the platform and the outer casing which will be around $40 and a stepper motor connected to the rotating disc/platform. An adequate model can be found for about $15 on hobbies stores for the stepper motor. It's of the type of bipolar rotating motor, using two-phase induction to provide a high enough torque for our needs. Also, this is the part of the final device that will be the most power consuming, the LASER and camera, and control module needing a much lower power input. The safe bet is then to choose a power supply that is the most adequate to our rotating device.

After some research, it is found that a 40watt, 12Volt classic power supply, of type PS1, is enough to fully power the stepper motor and have extra power for the rest of the components. It is again sold for about the same amount of $15 in multiple online stores. The next thing to think about is the LASER. One of the requirements are, like in most projects of this kind, a bright red LASER color. This makes the point-camera detection easier, both for initial accuracy and later for testing and calibrating the device. The wavelength for this would then be 650nm according to a standard spectrograph. The laser will be stationary so we will not worry about controlling or sending information for now, so the requirement will be stability and enough power. Knowing that a standard "fun" laser pointer is about 5 mw power, a 20 mw 650nm is a good choice for our application, powerful enough no matter what size our final model will be. LASERs of this caliber can be found online for about $30 for an industry-reliable model.

Next up is the camera. While we taught of using a regular webcam to have a quality product to provide reliability to our project. A more generic, hobby-oriented camera will provide more options when it comes to hands-on installing and testing. The extensively used OV7670 is a VGA camera that would do the job of detecting a red line in the dark, and for now the resolution is optimal. If we do well with this resolution in the testing phase, we may want a higher quality product and increase the resolution, and we would then think about getting a more sophisticated camera. But for our initial needs, this standard camera that we can acquire for about $10 online will be enough.

Additionally, we thought about adding another camera to improve efficiency of our module, and in this case we would just get another camera of the same type, but this again will be decided on a later stage of our design project. To control all of this, we have the choice between analog controls and digital. While we still haven't decided which one to use, the members experience in embedded systems will lead us to opt for a digital microcontroller for our project. We will need to control the mechanical rotating part, which should be an extensive job, as well as processing out images captured before sending them over to a laptop for calculations and 3D model creation. A medium scale microcontroller of the Atmel At mega type will be enough processing power and a good challenge to program. The microcontroller itself would be about $10, which would go into our custom PCB. We can get a good Printed circuit board for about $80 including shipping with companies like 4PCB.com.

Different circuit elements would be available through the various university's labs that we have access to, but we would allocate a variable amount, predicted at $40, for different parts of out circuit, like special capacitors, transistors, digital modules like flip-flops, voltage regulators and such elements. We also predict various hardware and connectors like USBs and wires to not exceed $20 in total cost. Last, we have the casing and esthetic elements Now for the parts that members most likely already have and do not require extra budgeting include a soldering iron and a dedicated laptop that runs MATLAB, a necessary program that will be extensively used for prototyping and testing as well as multiple steps of 3D model processing.

7.2.2 Other Expenses

Besides parts, others things that would require budgeting include software. While most of us have access to MATLAB, we will need extra programs for 3D modeling to take us from point clouds to 3D models. A safe bet allocation for purchasing these programs would be about $250, although we will try to find ways to minimize that. Another aspect to consider is training. These specialized programs might require dedicated and complicated ways of operation that could go beyond regular coding skills. These could also be just an accelerated way to master the programs to move on with optimizing our project. Training one or two members of the team can be very roughly estimated at $150.

7.3 Milestone

7.3.1 EEL 4914 – Senior Design I

Idea Finalization	05/23/2014
Initial Documentation	05/27/2014
Research Platforms	June
Stepper Motor Research	June
Camera Research	June

Laser/Controller Unit Research	June
Power Supply Research	June
Subsystem Integration	End of June
Design/Write	July
Finalize Design	2nd and 3rd Week of July
Paper Due	07/31/2014

Table 7.3 (1) Senior Design I Milestone Table

In the beginning of the term, we had several ideas on what project to do, we choose this project since it is something specific that has not been done and will boost our potential job careers since they will be put in the resume. We will explain what we did each one in this term like shown in the table above.

May – 2014: We started by making rough ideas on what the future project should look like, we spend 10 days deciding the topic of the project that we will be tackling. We had two ideas, one is to make a 3D scanner, and the other was to make a CPU from scratch, however, we chose the first one since we knew that the second choice was going to require a lot of more work and more money to put together. By the end of May, we knew exactly that we will be doing a short range 3D scanning device and we presented our main project in the initial project documentation.

June – 2014: By the first and second week of June we started gathering information from previous projects done in this matter, we discussed then in the research part of this thesis. Once, we had all the information, presented the initial project documentation, we started to learn more and more about all the specific parts and method of scanning 3D objects. We knew that this part of the project was going to be decisive in terms of how we will be tacking and constructing our project. By the third and fourth week, we completed the research part of this project, and started structuring our thesis, as well as knowing which team member what would be doing exactly in the next semester, we will discuss this in the following section

July – 2014: By the first and second week of this month, we started writing this documentation, by dividing our work, we wrote 30 pages each in research and design each one, it was one of the most difficult papers we had written in our whole career, it require a lot of instructions, which were well instructed, as well as methods, and formatting, we spent more than 3 days trying to come up with the best format of this project. By the third and fourth week of July, we pretty much finished typing this thesis which we accomplished successfully. Finally, we are planning to start working again on this thesis by the third week of August in the next semester in Senior Design II.

7.3.2 EEL 4915 – Senior Design II

Design Confirmation	07/31/2014
Order Parts	08/18/2014 – 09/01/2014
Build and Test Platform	09/01/2014 – 09/22/2014
Build and Test Stepper Motor	09/08/2014 – 09/29/2014
Install and Test Camera	09/15/2014 – 10/06/2014
Test Laser/Controller Unit	09/22/2014 – 10/13/2014
Install Power Supply	09/29/2014 – 10/20/2014
Subsystem Integration Testing	10/06/2014 – 10/27/2014
Testing/Debugging	10/27/2014 – 11/10/2014
Final Testing and Debugging	11/17/2014 – 11/19/2014
Presentation	11/20/2014

Table 7.3 (2) Senior Design II Milestone Table

August – 2014: By the third and fourth week of this month, we will be dealing with the confirmation of the design as well as ordering all the required hardware parts that we will be needing to build this system. We are actually planning to order dummy cheap parts to play with so that we know exactly which part will be using instead of buying the most expensive ones and resulting in not being able to actually integrating the system if some major problem might arise.

September – 2014: This month will be the most work intensive one, since we will plan on having all the parts ready and actually start constructing and prototyping this thesis. We plan on work on this project at least 50 hours per week in this month.

October – 2014: This month as well as September will be really work intensive, since we will have to decide finally which parts we will be using in our final design. It will require a lot of trial and error as well as coming up with clever prototype design.

November – 2014: We plan to start coding our project as well as debugging and testing. This is going to be the most difficult part in terms of getting that object scanned as well as represented in the software that we are planning to use.

December – 2014: Final month, we will be making final touches to our final product, and testing the final results, as well as presenting the thesis to a 3 staff committee. We will be working really hard in this semester.

7.4 Roles and Responsibilities

The roles and responsibilities were set up based on each individual skills and in the group's goals and desire for learning different technologies. While all members of the Short Range 3D Scanners project will help develop and design each aspect of the project, specific roles were assigned and accepted by each member to be the lead designer of their specific subsystem.

Nirav Kotadia – CpE: He will be working on the laser subsystem integration as well as the camera control part of this project. He is planning to start prototyping and buy parts by the end of August. Also, he will be in charge of helping with the programming part of the project since he has a pretty solid background in Java and C languages which will minimize the amount of work those other team members.

Franco Zavagnini – CpE: He will be in charge of the programming unit and algorithm design of this thesis, he will implement the laser triangulation as well as some of the controller part programming of the project. He has some modules and classes already designed, however, not actually implemented. He will require a lot of help from other team members since this a group project.

Gabriel Akenten – EE: He will be in charge of designing and building the power supply unit for the other components, he is an expert in power systems and has previous knowledge building power supplies as well as soldering and integrating complicated components electronically.

Naoufal Rihani – EE: He will be in charge of designing and programming the microcontroller as well as making sure that everything works perfectly in this matter. However, everyone will work with him, since integrating and programming the components is not something that only one person can do. He will be working along with Franco Zavagnini on the design and development of the programming part.

7.5 Safety

The prototypes that we will be developing for this thesis will be tested in a closed environment with minor outside light since we want to scan the object and model it in the data representation software as efficient as possible. Also, the testing environment we will try to keep it indoors since it can provoke damage to the laser sensor if placed outside where temperature and humidity might affect its correct functionality.

In terms of safety for end user, the only thing that can affect is the red laser since it can provoke eye damage to anyone staring at it directly for longer periods of time. We can use reflective glasses when manipulating the laser, especially in parts in which we will monitor of the whole scan. However, it is not something that is of main concern when handling this 3D scanning device. As a matter of fact, it can cause no harm to its user at all when handled properly. We are planning to enclose the whole circuit board and power supply unit so that it does not cause electrical short circuits if touched directly. We will try to keep every component secure so that we do not encounter any major issues when prototyping this thesis.

Finally, we are looking forward to take into consideration every secure aspect of this project so that it does not affect in any way its final users. Also, the testing environment will have to favorable for this to happen, and distance between laser and objects will be of great importance in this matter to not cause any major problem.

8.0 Conclusion

3D Scanners have been around for quite a few years and they still continue to be one of the most widely used devices for subsequent 3D printing. We will not get into details about 3D printing. However, it is something that we might consider in the future for future projects. Since we started this research, we have learned a lot about this technology which makes 3D modeling and rendering of object relatively easy and we will continue to strive for perfection as we go along this project. There are many 3D Scanning Devices out there in the market. The whole purpose of this project is to make one that is affordable, effective, with high performance, cheap, portable, not so heavy, and that it can get the job done meaning to scan and object in three dimensions, model it, and render it in a data representation software.

We have dealt and informed ourselves with several previous works in this matter and we have reached to the fact that to make an affordable 3D Scanning Device we have to constraint ourselves into making sure that the components that we buy will have all the reference material as well as the desired specifications to actually implement our design. We must not lose time dealing with wrong components since we have not that much time to accomplish our goal which is building our 3D Scanning device, coding our microcontrollers, testing, debugging, assembling, and presenting it. Basically, every 3D Scanner out there is expensive, big, not so portable, and they require more work and effort put to it. That is the main reason we chose this project as our final thesis. Also, every single one of the team members have skills that must be polished throughout this thesis. However, this does not mean that other team mates can comment about it. We are just trying to make the best possible within our reach 3D Scanning Device out there.

We are looking forward to start building our first prototype, and start coding along the way next semester; therefore, we designed something good and probably one of a kind 3D Scanner since it will be implemented with a series of algorithms developed by us and that will take a very important role in the final product. Also, there is still a lot to play around with, especially with lasers and microcontroller, our design is not definitive, it might have minor changes when we start building it.

We have learned a lot over the course of this semester, we put so much effort in designing and writing documentation. This was a team group project, everybody worked in their area, made research, came out with new ideas. One final thought to every adventurer out there that wants to start a project: Team work is absolutely necessary in these types of projects, without that, no project can be completed or even started. It requires a lot of sacrifice to make time to work on it, patience, and most important discipline in completing assignments and deadlines.

9.0 Bibliography

[1] NextEngine 3D Scanners, June 2014.(http://www.nextengine.com/assets/pdf/scanner-techspecs.pdf)

[2] Artec Spider specifications, June 2014.(http://www.artec3d.com/hardware/artec-spider/specifications/)

[3] MakerBotDigitizer_UserManual :page 64-66., June 2014.(https://s3.amazonaws.com/downloads-makerbot-com/digitizer/MakerBotDigitizer_UserManual.pdf)

[4] Flexible calibration Method for 3D Laser Scanner System: Yang Zhangdong , July 2014. (http://download.springer.com/static/pdf/805/art%253A10.1007%252Fs12209-014-2167-0.pdf?auth66=1405792466_f6f86d9617e8c94380e590decbab207b&ext=.pdf).

[5] Design and calibration of a hybrid computer vision and structured light 3D imaging system: Tom Botterill, Steven Mills, July 2014. (http://hilandtom.com/tombotterill/Botterill-Mills-Green-ICARA11.pdf)

[6] OptoNCDT Laser triangulation sensors, July 2014. (http://www.micro-epsilon.com/download/products/cat--optoNCDT--en-us.pdf)

[7] Laser Triangulation for shape acquisition in a 3D scanner: David Acosta, Olmer Garcia, July 2014.(http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=4019763)

[8] Stereo vision: Avidan,Pollefeys, July 2014.(http://webcourse.cs.technion.ac.il/236873/Winter2011-2012/ho/WCFiles/CV_stereo_2011-12_p.pdf)

[9] Stepper Motors (https://www.pololu.com/product/1205)

[10] Voltage Regulator (http://evergreenelectronics.in/lm317.html)

[11] D.C conversions (http://www.digikey.com/product-highlights/us/en/texas-instruments-simpleswitchers/4100, http://www.eidusa.com/Electronics_Kits.htm)

[12] The Digital Michelangelo Project: 3D Scanning. 2014 July. http://graphics.stanford.edu/papers/dmich-sig00/

Appendices

Appendix A

Table of Figures

Table of Tables

Appendix B

Permissions

```
the permission was hereby granted, provided that you add the correct
references to the image sources.

Best regards
Simon Winkelbach

--
Dr.-Ing. Simon Winkelbach
http://www.rob.cs.tu-bs.de/organization/staff/swi
Institut für Robotik und Prozessinformatik
Technische Universität Braunschweig
Muehlenpfordtstr. 23
38106 Braunschweig
```

Derrill Hartley July 2, 2014 6:52 PM
To: gabriel akenten
Re: Pololu - Stepper Motor: Bipolar, 200 Steps/Rev, 35x28mm, 10V, 0.5 A/Phase Inbox - iCloud

Hello, Gabriel.

You can use any of the images for the #1208 stepper motor from our website with proper attribution.

- Derrill

On Tuesday, 1 July 2014 at 3:45 PM, gabriel akenten wrote:

> Hello, Derrill
>
> Below is the link for the image and we want to use the image for our senior design project. We might order this motor during fall but for now we need an image for it to be approved.
>
> Thank you

Senior Design 1

Franco Zavagnini <francozb92@gmail.com>
para info

Hi, I am a CpE at UCF, I was wondering if I can get permission to use some figures on your website for my Senior Design project. Thanks

Senior Design 1 Permission

Franco Zavagnini <francozb92@gmail.com> 9:45 (hace 0 minutos)
para support

Hi, I am a CpE at UCF, I was wondering if I can get permission to use the figures on your website for my Senior Design project. Thanks

support@geomagic.com

Enviar

Anna Galdina Jun 19
To Me

Dear Nir

Thank you for asking the permission from us. Yes, you may use the picture of Spider with all rights reserved.

Best regards

--
Anna Galdina
Sales

Artec 3D
scanners

www.artec3d.com

.:oomlout:. II Aaron Nielsen June 30, 2014 4:21 AM
To: gabriel akenten
Re: SERV-03-MI (Micro Servo) Inbox - iCloud

Hi Gabriel;

Not a problem. Please feel free, could you send us a link to your project once it's online. (or don't worry if it is not going to be published online).

Regards
Aaron Nielsen
aaron@oomlout.com

See More from gabriel akenten

gabriel akenten June 29, 2014 8:52 PM
To: info@oomlout.com
SERV-03-MI (Micro Servo) 1

So we are looking forward in using an image of the servo(SERV-03-MI (Micro Servo)) in our project.
Thank you