Palestine Polytechnic University

College of Engineering

Graduation Project

Automated Linear Scanning Book System

By

Mohammad Hasan Ismail **Mohammad Ahmad Shakhtour**

Ahmad Hasan Dadou

Supervisor:

Dr. Osama Ata

Submitted to the College of Engineering

in partial fulfillment of the requirements for the

Bachelor degree in Mechatronics Engineering

Hebron, May 2017

Palestine Polytechnic University

Collage of Engineering

Mechanical Engineering Department

Hebron – Palestine

Automated Linear Scanning Book System

Project Team:

Ahmad Hasan Dadou

Mohammad Ahmad Shakhtour

Mohammad Hasan Ismail

Submitted to the Collage of Engineering

In partial fulfillment of the requirements for the

Bachelor degree in Mechatronics Engineering.

Supervisor Signature

....

Testing Committee Signature

....

Chair of the Department Signature

....

May 2017

Abstract

Digitizing books and documents has become increasingly prevalent amongst libraries and university facilities as large data storage becomes cheaper and more available and physical storage space becomes scarce. There are currently many commercial scanning methods available, but all have trade-offs between speed, reliability, automatic page turning and price.

In this project, we proposed the design of an automated linear scanning book system which has several attributes. For a start, the moving saddle by two stepper motors can hold up to 3 Kg. The system contains a suitably designed channel slot, this design has implemented a new variation on the idea of pneumatic page turning. This involves the use of a vacuum cleaner device to pull book pages into a channel that flips them to the other side of the book, in order to allow two installed scanners to take an instant image of the two exposed pages, to use them at a later stage to convert an merge them into a PDF file.

The material choice of the system housing was Acrylic. This is because it is easier to laser cut and fabricate, relatively lighter than Aluminum, wood and stainless steel. The interconnection performance is based on an Arduino Uno and a Raspberry Pi microcontrollers for controlling the motors, scanners, sensors, switches, vacuum device, in addition to transmitting the images to a desktop or laptop. The overall system operation performance is very satisfactory. However, we recommended a scanner with faster speed, a vacuum device with lower noise. However these were the only available in local markets.

Apart for these recommendations, our system is low priced, reliable in page flipping and image storing. Moreover mechatronics design procedure steps have followed in the implementation of the system.

ملخص المشروع

الكتب والوثائق الالكترونية أصبحت أكثر انتشارا في مرافق الجامعة ، حيث أصبح بالإمكان تخزين بيانات كبيرة بإمكانيات صغيرة واقل تكلفة وسهولة الوصول إليها، أما طرق التخزين التقليدية أصبحت تشكل عبئا من ناحية التخزين والوصول إليها.

يوجد في السوق العديد من الأجهزة التي تعمل على تحويل الكتب المادية إلى كتب الكترونية ولكن تعتبر باهظة الثمن بالنسبة للمكتبات الصغيرة ذات الإمكانيات المحدودة، و للحد من هذه المشاكل سنعمل على تصميم نظام اتوماتيكي لتحويل الكتب المادية إلى نسخ الكترونية ، بحيث يعتمد هذا النظام على الهواء في عملية سحب الورقة إلى مجرى خاص يسمح بقلب الورق إلى الجهة الثانية من الكتاب .يعمل النظام على مسح للكتب المجلدة وتحويلها الى نسخ الكترونية وتخزينها بصيغة PDF .

Table of contents:

Lists of Tables:

Lists of Figures:

1

Chapter One

Introduction

1.1 Overview.

1.2 Objectives.

1.3 Project Importance.

1.4 Requirements

1.5 Motivation

1.6 Project schedule and time plan.

1.7 Budget.

1.1 Overview:

This chapter presents, a general description of the system, starting with a background, objectives, importance of the system, requirements, motivation, project schedule and time plan and budget.

1.2 Objectives:

The main objectives that we are looking forward to achieve in our project are the following:

1) Design, construct and test the system prototype.
2) Utilize a microcontroller (Arduino) and build a code to control the operation of system units.
3) Build the scanning system to accommodate books of different sizes ,operate with an acceptable noise level, provide good image quality, and dependably turn each page without damage
4) Reduce the cost of manually converting a whole book into softcopy.
5) Making the system user friendly.
6) To achieve high quality PDF file.

1.3 Project Importance:

The importance of the project is summarized by the following points:-

- Scanning devices have become indispensable to students, teachers and administrators at universities and academic institutes, in particular. However the problem remains when it comes to scanning books with binders. Here the direct human interface becomes necessary and the whole process takes time and effort.
- An automated process would be required if human interface is to be avoided which would make scanning effortless, less time consuming and without possible damaging.

1.4 Requirements:

This system requires hardware and software components in order to make a prototype for the system:

1.4.1 Hardware requirements:

- Microcontroller
- Raspberry PI
- Stepper Motor
- Vacuum
- Sensors
- Scanners

1.4.2 Software requirements:

- Programming languages for Raspberry Pi such as Python
- Arduino programming language C

1.5 Motivation:

- All the modern scanning devices have not the ability for automatic scanning of bound books, these devices need the human interface to flipping the pages by hand and this process consume time and effort.
- The system is an implementation that contain a mechanical design and electrical design and require an integration of these designs to achieve the objects of this system, so this is the essence of mechatronics engineering.

1.6 Project schedule and time plan

Table 1.1: Tasks description.

	Task description
T1	Project Selection
T2	Collecting References from libraries
T3	Collecting References from websites
T4	Select an initial design
T5	Create and draw the selected design on SolidWorks
T6	Do some adjustments on the design
T7	Select the mechanical parts and sensors
T8	Writing the text
T9	Prepare the 1st presentation
T10	Make the required adjustments on the introduction text
T11	Buy the mechanical and electronic parts
T12	Do some variations if required
T13	Build the project
T14	Put the mechanical and electronic parts of the system
T15	Build the suitable code for the system
T16	Test the result
T17	Do some justification to make the system more efficient
T18	Try to correct the expected holes of the system
T19	Writing the text
T20	Make a final adjustments on the text
T21	Prepare for the final presentation
T22	Make more test on the system

Table 1.2: First semester time table

Task/Week	1	2	3	4	5	6	7	8	9	10	11	12	13	14
T1														
T2														
T3														
T4														
T5														
T6														
T7														
T8														
T9														
T10														

Table 1.3: Second semester time table.

Task/Week	1	2	3	4	5	6	7	8	9	10	11	12	13	14
T11														
T12														
T13														
T14														
T15														
T16														
T17														
T18														
T19														
T20														
T21														
T22														

1.7 Budget

When you going to make any project, you have to take attention that your project will cost a certain amount of money. If you plan to make a development for your project, you must take attention more and more, to minimize the cost as possible as you can, to make a beautiful profit margin. Table 1.4 shows the budget of the system.

Table 1.4: Budget

No.	Component name	Price
2	Stepper Motor and driver	150 $
2	Scanners	200$
1	Vacuum Cleaner	50 $
1	Arduino Uno	20$
1	Raspberry Pi	150$
-	Acrylic Sheets “3 square meter” and machining	380$
-	Screws ,nuts ,belt and pulleys	30$
-	Sensors (Limit switch & Reflective Optical Sensor) and wires	80$
1	Power supply	60$
	Total	1100 $

2

Chapter Two

Conceptual Design and Functional Specifications

2.1 Overview.

2.2 State of the Art

2.3 Needs and requirements

2.4 Proposed Designs and Evaluation.

2.5 Conceptual Design.

2.6 Functional specifications.

2.7 Block diagram and Flow Chart

2.1 Overview:

This chapter will show some suggested designs for the project, and describe the conceptual design of the project, depend on the requirements and objectives that have been clarified in the previous chapter, and it is describe the functional specifications of the components and the relations between them. And there is a flowchart that clarify the steps of the system operation and how to control the different components to make the system working well.

2.2 State of the Art

The process of scanning books, articles, encyclopedias and different types of documents become very important for our life these days. Libraries have traditionally been very limited in the individuals that they can reach, space for texts, the ability to search for specific information, and other factors. By storing the information in books digitally, millions of texts can be easily kept in a small space for a low cost without degradation, and can be made available to anyone at any time over the internet. In order to fill these digital libraries, books must be scanned into databases on a massive scale. This “mass digitization” aims not only to make collections of books available electronically, but rather to make every book ever printed available on a computer. So many devices and systems exists to achieve the aims of this process. In the following part a summary of some devices and systems that help for the process of scanning.

2.2.1 Scanner

A scanner is a device that captures images from photographic prints, posters, magazine pages, and similar sources for computer editing and display. Scanners can be used to convert images or text on page into a digital format that can be used by the computer [1].

A scanner works by shining a beam of light onto the surface of the object that you are scanning. This light is then reflected back onto a sensor that detects the color of the light. This is then used to build up the digital image.

Items that are scanned are usually stored in an image format. However, special software - Optical Character Recognition - can be used to convert text on the page into text which can be edited with a word processor. However, the text doesn't always get converted very well and you could end up with a lot of mistakes.

There are two types of scanner:

- Flatbed scanners
- Handheld scanners

The most popular type is the flatbed scanner. This is shown in **Figure2.1** that has its lid raised to show the glass surface where you place the item to be scanned. This is probably the one that you use at school. They can scan larger images and are more accurate than handheld scanners.

Handheld scanner which is shown in **Figure2.2** is usually only a few inches wide and are held in the hand whilst they are rolled across the document to be scanned. The images produced are generally not as large or as high quality as those captured with a flatbed scanner.

Both of Flatbed and Handheld scanners are need the direct human interface for scanning process especially for flipping the pages of books with bound ,this is consume time and efforts [1].

Figure 2.1: Flatbed scanner

Figure 2.2: Handheld scanner

1.2.2 Cardboard Scanner

The cheapest option is to build your scanner from a cardboard box which is shown in **Figure 2.3**. It is also a relatively easy rig to build. The quality may suffer, but in some situations that it a good trade-off to make. This design contain a camera to capture an images of the book ,the process of this design is slow ,need human interface to flip the pages of books and the image quality is low[2].

Figure 2.3: Cardboard Scanner

2.2.3 Hardware Store Scanner

The hardware store scanner which is shown in **Figure2.4** is a more durable alternative to the cardboard scanner. No custom parts are needed and the only tools you need are standard woodworking power tools. If you walk into a hardware store, you can walk out with everything you need to build this scanner. You can expect to get better scans with this rig than the cardboard scanner [3].

Figure 2.4: Hardware store Scanner

2.2.4 Archivist Quill Book Scanner

Built from aluminum, steel, and plastic as shown in the **Figure 2.5** , the Archivist Quill book scanner provides a high quality book scanning appliance. The base is designed to provide an ideal photographic environment for scanning books. A pair of Canon PowerShot ELPH 160 cameras capture 20 MP photographs. The system is controlled by a Raspberry Pi [4].

The Archivist Quill maximizes the quality of the images. A V-shaped cradle and platen system presses the pages flat against glass while being gentle on book spines. High quality LEDs provide bright and even light from above. The geometry of the camera and lights is designed to remove nearly all glare. Potentially reflective surfaces on the machine are black to avoid lighting artifacts.

High quality book scanning no longer requires the resources of a corporation or large institution. The Archivist Quill is cheap enough that individuals and small organizations can afford it. And it is compact and light enough to be shipped easily anywhere in the world.

But the process of flipping the pages is still depends on direct human interface, so this device consume time and efforts.

Figure 2.5 Archivist Quill Book Scanner

2.2.5 ScanRobot 2.0 MDS

Figure 2.6 shows an automatic book scanner produced by Treventus, uses suction to pull book pages upright while scanning vertically, and is reported to be able to scan 2,500 pages per hour. While it is very quick, it is usually priced around $95,000 [5].

Figure2.6 ScanRobot 2.0 MDS

2.2.6 DL3000:

A commercially available book scanner is the DL3000 from 4DigitalBooks as shown in **Figure 2.7.** This is an automated book scanner that is extremely fast. With this scanner, the books lie flat, then suction lifts a page and a flat piece is inserted below it to turn the page. There is also a mechanism to raise and lower the book on each pass so the pages do not get bend or crimp. Although this machine is quick, reliable and fully automated, the cost is significant at about $250,000 [6].

Figure 2.7: DL3000

2.2.7: Linear Book Scanner

Because most of the commercially available devices can range from $50,000 to $250,000, they are inaccessible to smaller libraries. The ultimate goal of this project is to build a book scanner out of readily available parts for under $1500. **Figure 2.8** shows a prototype produced by Dany Qumsiyeh [7] accomplishes this goal but does not produce the quality or reliability of a fully functional machine. The ultimate goal is for the scanner to be able to accommodate books of any size, so our task is to improve the size and shape of the channel so that a larger range of pages can slide easily through it without ripping or bending. Another significant problem presented was reducing the cost. As previously mentioned, the linear book scanners used currently in libraries can range from $50,000 to $250,000 for a single scanner. Ideally, we would like to make this technology accessible to more people by building a machine that is affordable to smaller libraries and collections of work in the developing world. The prototypes built by Dany cost less than $1500, and one of our goals is to continue to reduce the cost and improve the quality of the new device (the system). This design are clearly not portable and are very difficult to move even from room to room. Additionally, it can have complicated user interfaces, while the new prototype, should be user-friendly, so anyone would be able to use it instead of a trained technician [7].

Figure2.8 Linear book Scanner

2.3 Needs and Requirements

1) The system should be able to scan 300 page per hour.

2) The cost of the system should be less than 1500 $.

3) The system can scan different sizes of books (the maximum size of the page is A4).

4) The system can scan books with weight 3kg and less.

5) The power feeding the system must be 220 V.

6) Ergonomic Design.

7) The system should be easy to use and safe.

2.4 Proposed Designs and Evaluation

After identifying the project idea, there is a different proposed designs for the system, after making an evaluation the different designs depends on the objectives and requirements, then the option is going to select the more effective design.

2.4.1 Design No.1

This design as shown in **figure2.9** assuming that the book is placed on a fixed position on the device, and the process of flipping the pages depends on a mechanical arm. The scanner moves along the book and takes the pages photo. But after we evaluate it, we visualize that the process of controlling the mechanical arm is difficult and the expected results will be not efficient and more expensive than the other designs.

.

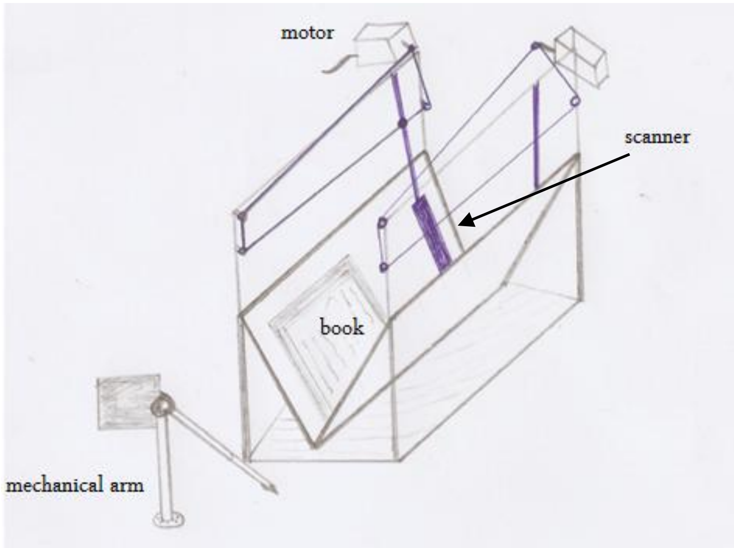


Figure 2.9: Design No.1

2.4.2 Design No.2

This design as shown in **figure 2.10** is inspired from previous project "Linear book Scanner", the design consists of a moving "saddle" seated on an inverted V-shaped base that forms the body of the scanner. The process of scanning the pages depend on the movement of the saddle, in this design the scanning process occurs in the forward movement ,and the process of flipping the pages occurs in the backward movement ,so this design consume more time than the final design.

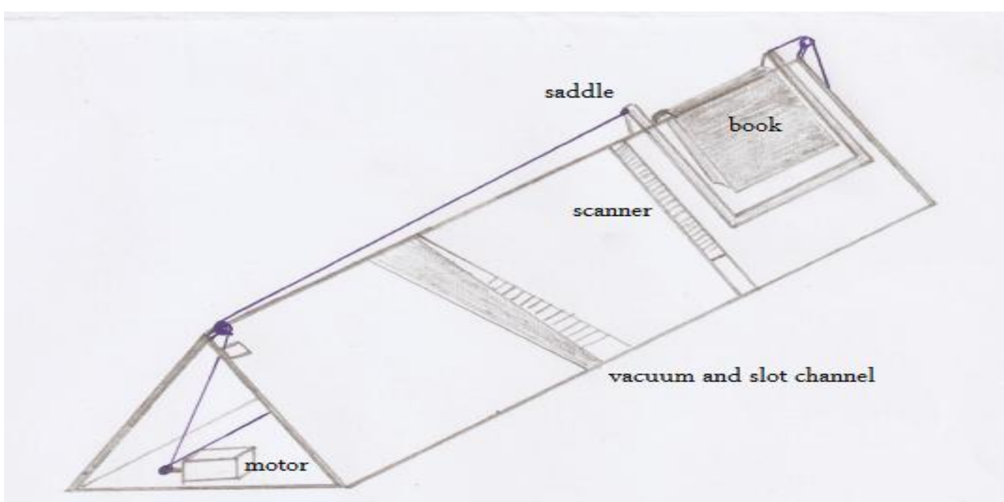


Figure 2.10: Design No.2

2.4.3 Design No.3 (Final design)

This design as shown in **figure 2.11** is an improve from the design No.2, the design consists of a moving “saddle” seated on an inverted V-shaped base that forms the body of the scanner. The process of scanning the pages depend on the movement of the saddle, in this design the scanning process occurs in the forward movement ,also the process of flipping the pages occurs in the forward movement ,so this design consume less time than the previous design. And less expensive than the design No.1.

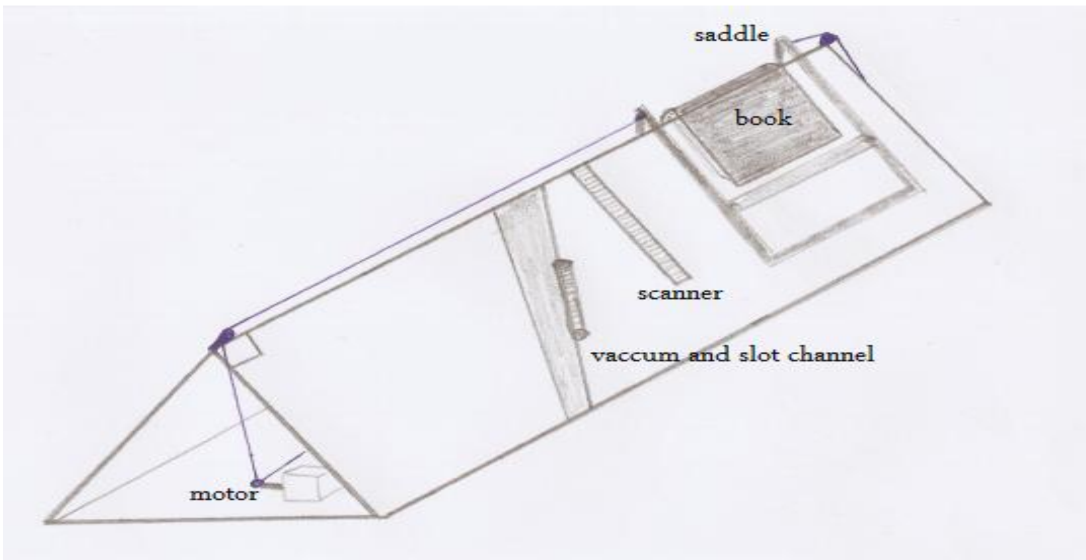


Figure 2.11: Design No.3 (Final design)

2.5 Conceptual Design

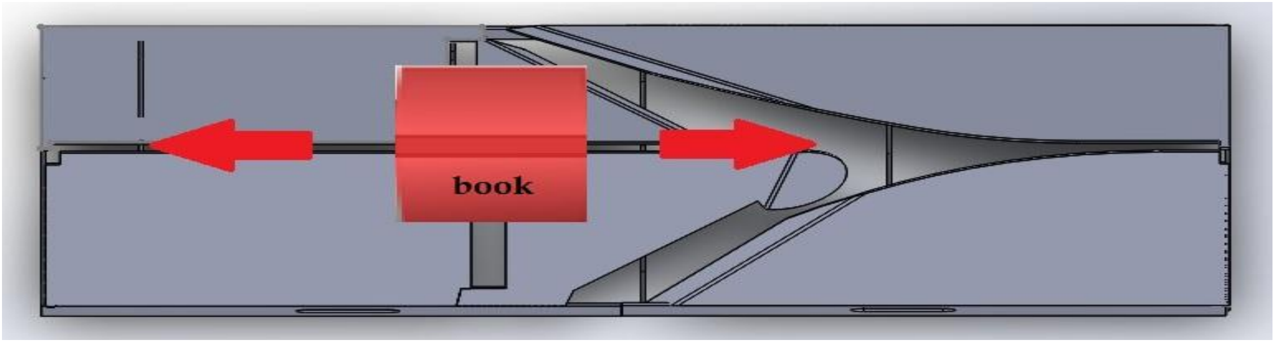
After defining needs and requirements, and generate some solution by brain storming and morphological chart. Now we want to specify design concept that meet project requirement and functional specification to choose candidate design, to move forward to functional specifications in next section.

The purpose of the conceptual design is to define required parts and how it will work together, and the relationships between the components.

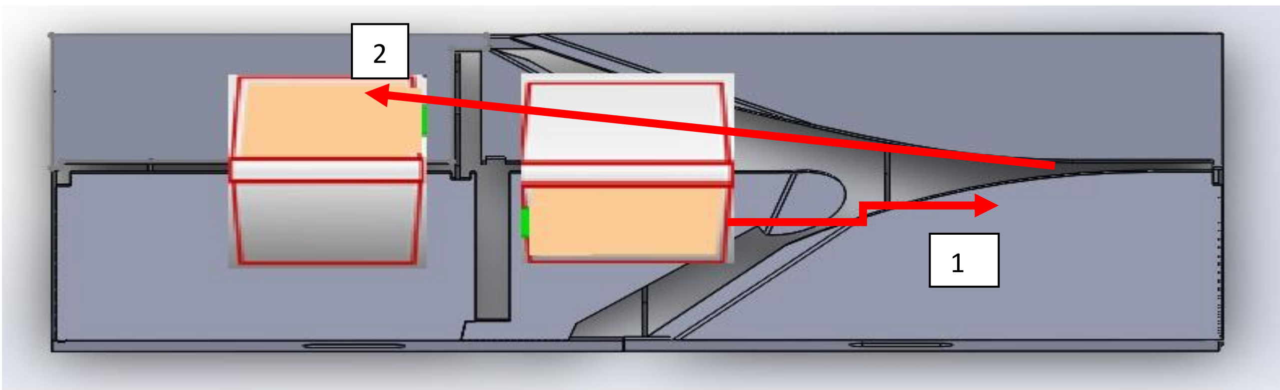
The system consists of a moving “saddle” seated on an inverted V-shaped base that forms the body of the scanner. A channel is cut through the top of the body to allow pages to fall through it [**Figure2.12 a-c].** To make a scan, the user places a book face-down in the saddle and presses a button to begin the automated process.

A motor moves the saddle down the body of the machine and across two digital scanners that create images of the exposed pages. The book continues to move along the body until it reaches a vacuum that “grabs” the page that is currently facing down and turns it so that it can be scanned. The book still moves along the body, but the “grabbed” page is funneled into the central channel. Once the entire length of the page has been fed into this channel, the motor reverses direction to return the book to its starting position. As it moves backwards, the page in the vertical channel is funneled onto the opposite side of the body, effectively turning the page. Once this has been completed and the book returns to its original position, and the process begins again. The operation cycle will continue until scanning all the pages of the book.

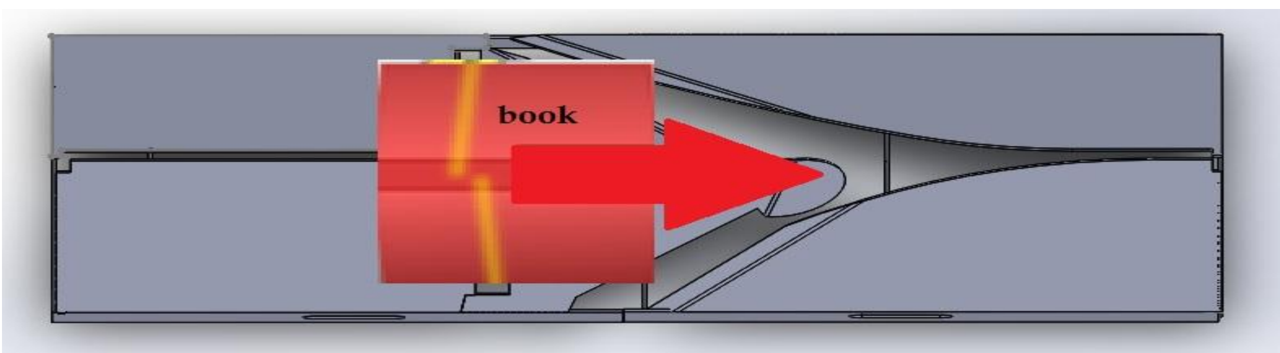
Figure 2.12A shows that the book moves back and forth over the machine, **Figure 2.12B** shows a book mounted on a saddle, and the slot channel which a page is absorbed by the air of the vacuum to the center of the system this is shown in position **1** and the page move to the other side thorough backward movement this is shown in position **2**. **Figure2.12C** shows the process of scanning the pages after the page is flipped through the slot.



A) A book moves back and forth over the machine



B): A vacuum sucks a page from one side to the other



C): The pages are scanned as they travel across two imaging sensors

Figure2.12: Steps of the system operation

2.6 Functional specifications

It explains how everything works together in greater detail and compliant with requirements, and exploring the feasibility of a product, to achieve the most valuable design.

1) Using the Acrylic sheet in frame building, because the Acrylic is a relatively soft, durable, lightweight, and easy to machine by laser cutter.
2) Using two scanner for the process of scanning the pages.
3) Using vacuum for sucking the pages.
4) Using Arduino to control the stepper motor, vacuum and sensors.
5) Using a Raspberry Pi to connect the scanner, Arduino and storing the pages
6) Using limit switch to identify the start of the path.
7) Two stepper motors to move the saddle.
8) Ultrasonic sensor to check the number of pages that the vacuum sucking it.
9) The shape of the system is prismatic and the dimensions are **(length 120 cm, height 30 cm, width 40 cm)**

2.7 Block diagram and Flow Chart:

Figure 2.13 shows block diagram that describe the system components and how it is connected together, to integrate the system and make it work well.

Block diagram shows the basic components, which are the Raspberry Pi and the Arduino, the Arduino controls the operation of stepper motors and the vacuum and the other sensors. Arduino connects with the Raspberry Pi. Raspberry Pi will operate the scanners and mange the image processing.

Figure 2.14 shows the flow chart and describe the algorithm of the system. The system start when someone push on a start button, then the stepper motor start work in the first direction, the saddle will move until reach the scanner, the speed of stepper motor slow down to be suitable with the scanning process and the scanners take images of the pages, when the saddle reach the slot channel, the vacuum device turn on for a specific time to suck a page, the ultrasonic sensor checks the number of page, if there is a problem in the number of pages in the slot channel the system will stop, else the saddle still move until reach the end, then the stepper motor will change the direction of movement. The saddle will return back, as it moves backwards, the page in the vertical channel is funneled onto the opposite side of the body, when the saddle reach the start point the process will return again.

By using number of steps depends in the system length, the system identify the locations of scanners, vacuum and sensor of pages.

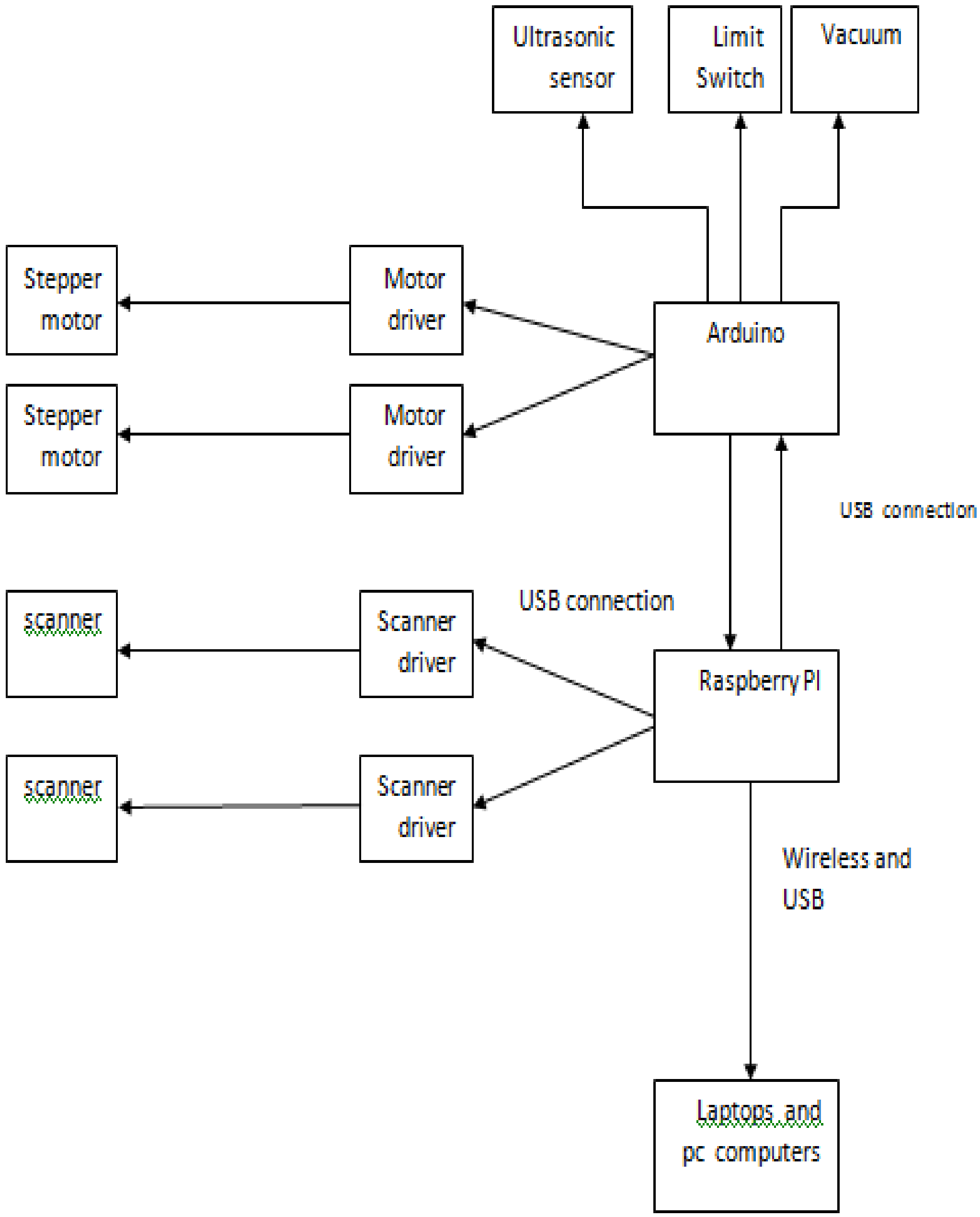


Figure2.13: Block diagram of the system

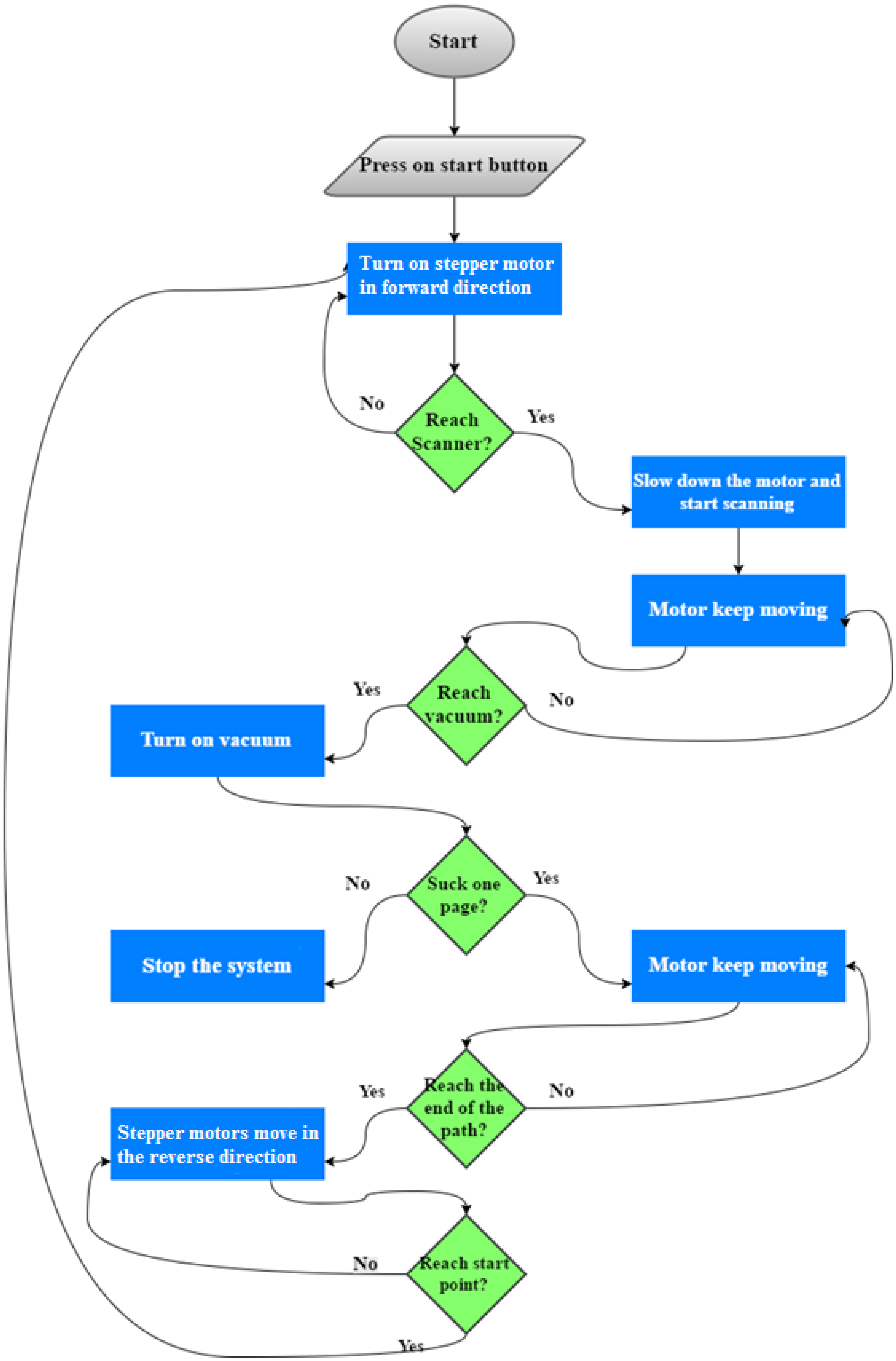


Figure2.14 Flowchart of the system

3

Chapter Three

Calculations and Components Selection

3.1 Overview.

3.2 Calculation of the system

3.2 Stepper motor.

3.3 Scanner.

3.4 Arduino.

3.5 Vacuum Device.

3.6 Raspberry Pi.

3.7 Sensors.

3.8 System switches

3.1 Overview

This chapter clarifies the desired calculation and explains the components of the system, principle of operation, specification and usage. All components should be well integrated into the system.

3.2 Calculations

3.2.1 Speed of System

Depend on the need and requirements (300 page per hours) the system speed should be as follow

- 300 page/hour that means 150 cycle of the system because the system scan 2 pages in every cycle
- The system needs 1 hour for 150 cycle that means one cycle need 24 seconds
- Depend on the speed of system, every cycle needs 24 seconds, the 24 second will divided to the scanning process and the process of the saddle movement, suppose the scanner needs 14second and the rest of cycle needs another 10 seconds.
- Speed in the scanning process =distance /time

  =30/14 =2.1cm/sec
- Speed in the rest process =108/10 = 10 cm/sec

3.2.2 Motor Analysis

It is important to understand the dynamic loads on the motor during operation to know how large the motor should be. To determine the required torque, estimates of the height, acceleration, and friction of the saddle and book were made. These were used to determine the maximum torque on the motor.

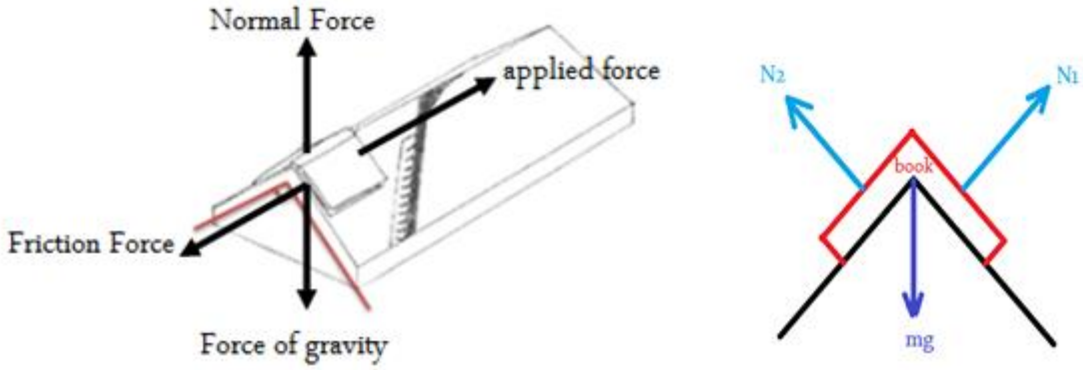


Figure 3.2 Free body diagram of the system

From **Figure3.2** the frictional force was found to be

We suppose N1 =N2 =N (the book is open from the half)

$$(N1 + N2)\sin\frac{\theta}{2} = mg \tag{3.1}$$

$$N = \frac{mg}{2\sin\frac{\theta}{2}} \tag{3.2}$$

$$F_f = 2\mu N \tag{3.3}$$

$$F_f = \frac{\mu mg}{\sin\frac{\theta}{2}} \tag{3.4}$$

Where:

N: Normal force

μ: the coefficient of friction at a maximum of 0.3

m:the book mass of 3.5 kg,

g: the acceleration due to gravity

θ: the angle of the book scanner (seen as a in the diagram) of 90° .

F_f: the friction force equals14.5 N.

And the acceleration force was found to be

$$a = \frac{Vmax^2}{2d} \tag{3.5}$$

$$F_a = ma \tag{3.6}$$

$$F_{total} = F_a + F_f \quad (3.7)$$

$$RPM\ max = \frac{Vmax}{R} \times \frac{60}{2\pi}$$

Where:

d: the distance of acceleration equals 0.03 m

Vmax: suppose the maximum velocity of the system equals 0.5 m/s

a: the acceleration equals 4.1 m/s^2(by calculation)

Fa:the acceleration force = 14.5 N (by calculation)

R: radius of the pulley equal 6.4 mm

This gives a total force of 29N. A safety factor of 2 was added to account for these things and brought the total force up to58 N. And the maximum speed 750 rpm

$$T = F_{total} \times R \quad \textbf{(3.8)}$$

where:

T : is the torque

so the torque equal 37 N.cm or 3.8 Kg.cm. NEMA 23[1] stepper motor was chosen for this task.

3.3 Stepper motor

A stepper motor is a brushless DC electric motor that divides a full rotation into a number of equal steps. The motor's position can then be commanded to move and hold at one of these steps without any feedback sensor (an open-loop controller), as long as the motor is carefully sized to the application in respect to torque and speed.Controlling the stepper motor speed can be by using Arduino, and making it variable depending on scanner and vacuum operations.

[1] Appendix A

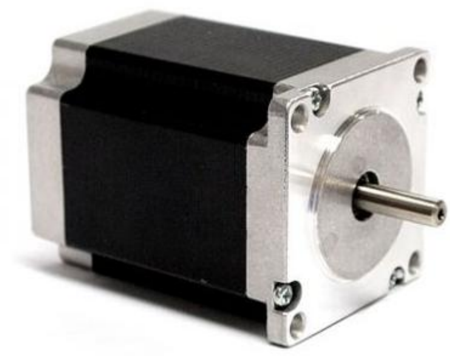


Figure3.3 Nema 23 stepper motor

Specifications:

1) Size: 56.4 mm square × 76 mm, not including the shaft (NEMA 23)
2) Weight: 0.7 kg
3) Shaft diameter: 6.35 mm (0.25″) “D”
4) Steps per revolution: 200, step equal 1.8 degree.
5) Current rating: 2.8 A per coil
6) Voltage rating: 3.6 V
7) Resistance: 1.13 Ω per coil
8) Holding torque: 90 N.cm (max)
9) Inductance: 3.6 mH per coil
10) Lead length: 30 cm (12″)
11) Output shaft supported by two ball bearings

3.4 Scanner

A scanner is a device that captures images from photographic prints, posters, magazine pages, and similar sources for computer editing and display. Scanners come in hand-held, feed-in, and flatbed types and for scanning black-and-white only or color. Very high resolution scanners are used for scanning for high-resolution printing, but lower resolution scanners are adequate for capturing images for computer display [1].

The most important aspect of the scanner is that it needs to be SANE (Scanner Access Now Easy) compatible. SANE is a set of open source drivers for Linux that allows scanners to be operated from the command line. This is critical to be able to quickly write a script that coordinates the scanning to the book movement. Canons line of LIDE scanners are some of the few scanners currently produced that are fully SANE compatible.

Depends on the speed of the system as mentioned in the requirements, the scanning process needs 13 seconds. The best choice for selecting scanner is product from Canon company "Canoscan LIDE 220" [2] which is shown in **Figure 3.4.** It has high speed scan, high resolution, good size for frame design and easy to control by Raspberry Pi. The system needs two scanners for both sides. But this type doesn't available in Palestine.

Figure 3.4 Canoscan LIDE 220

By the way there is another product available in the local market, so the choice is going for it. Which is a Canoscan Lide 120, this product is slower than Canoscan Lide 220, but can do the task of scanning process.

Table 3.1 shows the differences between the two model.

[2] Appendix A

Table3.1 Specifications of scanners

Specifications	**CanoscanLide 220**	**CanoscanLide 120**
Scanner type	Flatbed	Flatbed
Maximum document size	A4/letter (216x297mm)	A4/letter (216x297mm)
Scanning speed (color):	2.2msec/line (300 dpi) 33.2 msec/line (4800dpi)	4.4msec/line (300 dpi) 55.2 msec/line (4800dpi)
Scanning speed (grey scale):	2.2 msec./line (300 dpi), 11.1 msec/line(4800dpi)	4.4 msec./line (300 dpi), 20.2 msec/line(4800dpi)
Power supply	supplied via USB port	supplied via USB port
Supported operating system	Windows, Linux ,Mac	Windows, Linux ,Mac

3.5Arduino

Arduino is an open-source microcontroller. Hence one can easily program it in (C++). Arduino provides high speed transfer of information to the stepper motors. It is also used to control the system parts which includes sensors, vacuum and the stepper motors movement and speed [10].

Figure3.5: Arduino Uno

Arduino specifications:

1) Microcontroller : ATmega328
2) Operating voltage : 5 V
3) Digital I/O Pins : 14 (6 provide PWM)

4) Analog input pins : 6
5) DC current per I/O pins : 40 mA
6) Dc current for 3.3 V pin : 50 mA
7) Flash memory : 32 KB
8) SRAM :2 KB
9) Clock speed : 16 MHz

3.6 Vacuum Cleaner:

A Vacuum cleaner is used in this system to help in flipping the book pages. It should have enough force to turn a page and can be considered as a primary concern when choosing an appropriate device tube. Vacuum choice will depend on the tests. Small hand-held vacuum is suggested as a first choice but it fails on the test, the second choice is a vacuum cleaner which have specifications better than the first one, and it would be sufficient to turn the pages.

Vacuum cleaner specifications:

1) Power supply: 220 volt Ac
2) Input power: 2000 W.
3) Airflow: 35 L/sec
4) Vacuum pressure: 30 KPa.
5) It has a potentiometer that control the airflow, which helps the system to change it as needs.

3.7 Raspberry Pi

The Raspberry Pi is a low cost, credit –card sized computer that plugs into a computer monitor or TV, and uses a standard keyboard and mouse. It is a capable little device that enables people of all ages to explore computing, and to learn how to program in languages like Scratch and

Python. It is capable of doing everything like a desktop computer, from browsing the internet to making spreadsheets, word processing, image processing and playing games [11].

The Raspberry Pi is a small, barebones computer developed by the Raspberry Pi Foundation, a UK charity, with the intention of providing low –cost computers and free software to students. Their ultimate goal is to foster computer science education and they hope that this small, affordable computer will be a tool that enables that.

Figure3.6 Raspberry Pi

Raspberry Pi 3 specifications:

1) CPU: Quad cortex A53 @ 1.2GHz.
2) GPU: 400 MHz VideoCore IV.
3) RAM: 1 GB SDRAM.
4) Storage: micro-SD.
5) Ethernet: 10/100.
6) Wireless: 802.11n/Bluetooth 4.0.
7) Video output: HDMI / composite.
8) Audio output: HDMI / Headphone.
9) GPIO: 40.

The purpose of using Raspberry Pi 3 is to allow the system to operate with the scanners and the Arduino microcontroller. It is also used in the process of converting the images into a PDF file.

3.8Sensors

3.8.1 Limit Switch

In this work the limit switch which is shown in **figure3.7** is used to detect if the saddle is at the start position (Homing).

Figure3.7 Limit switch

3.8.2 Reflective Optical Sensor (Infrared Sensor)

Figure 3.8 shows the TCRT5000, which are reflective sensors which include an infrared emitter and phototransistor in a leaded package which blocks visible light.

The system will use reflective optical sensor to detect if the page is exist or not. This will solve the problem of missed scanned pages, by detecting pages in turning operation and providing the microcontroller a command to stop the device and return the missed scanned pages to scan again [13].

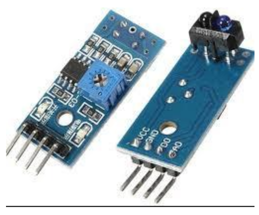


Figure3.8TCRT5000 Sensor

3.9 System Switches

3.9.1 On/Off Switch

Figure 3.9 shows the switch that have been used in the system. Switch is an electrical component that can "make" or "break" an electrical circuit, interrupting the current or diverting it from one conductor to another, in the system there is two switches, one for reset and the other is for turn on the system or turn it off.

Figure3.9: On/Off Switch

3.9.2 Emergency switch

Figure 3.11 shows the emergency switch. An Emergency switch is defined as a fail-safe control switch or circuit that, when de-energized, will stop the operation of associated equipment and will shut off all potential hazards outside the main power enclosure. Emergency switch, or "E-Stops", are a special type of pilot device that perform the emergency shutdown operation on a machine or electrical system. In this project emergency switch is needed for the state that have some problems.

Figure3.11: Emergency switch

4

Chapter Four

Mechanical and Electrical design

4.1 Overview.

4.2 Engineering Design Process.

4.3Main frame principle design

4.4 Material Selection.

4.5 Forces and stress analysis

4.6 Final Design

4.7 Electrical design

4.1 Overview

After defining the problem, needs, the conceptual design for the project and project requirements in the previous chapters, now this chapter explains:

1) How to design the frame and discuss the foreseen problems in the design and how to solve it.
2) Select the optimal solution and explain why should be selected.
3) Material selection.
4) Forces analysis.
5) Determine the dimensions depending on many considerations such as strength, reliability, appearance, friction, processing ,cost, safety, weight, noise
6) The electrical design.

4.2 Engineering Design Process.

The engineering design process is a formulation of a plan or scheme to assist an engineer in creating a product, and it also defined as the process of devising a system, component, or process to meet desired needs. It is a decision making process (often iterative) in which the basic sciences, mathematics, and engineering sciences are applied to convert resources optimally to meet a stated objective. Among the fundamental elements of the design process are the establishment of objectives and criteria, synthesis, analysis, construction, testing and evaluation.

There are five basic steps process usually used in a problem-solving works for design problems as well. Since design problems are usually defined more vaguely and have a multitude of correct answers, the process may require backtracking and iteration. Solving a design problem is a contingent process and the solution is subject to unforeseen complications and changes as it develops.

The five steps used for solving design problems are:

1) Define the problem
2) Gather pertinent information
3) Generate multiple solutions
4) Analyze and select a solution
5) Test and implement the solution

4.3 Main Frame Principle Design

4.3.1Frame design

The manual method to scan a book is to put the book at desired pages and then the scanner move and scan the pages, then flipping pages done manually and scanning process starts again, depending on this principle, in the design of the system the operation of scanning a book and flipping the pages is done automatically.

The final conceptual designassume that the book moves on a saddle and the scanner is fixed, unlike the usually book scanners in the markets, in order to use book movement in flipping pages and to get high performance in scanning operation, the proposed frame style of the system as Triangular Prism with 90° as shown in **Figure 4.1** .The choice of Triangular Prism is to suit with the book shape and movement, the purpose of our choice is to get high performance in scanning operation.

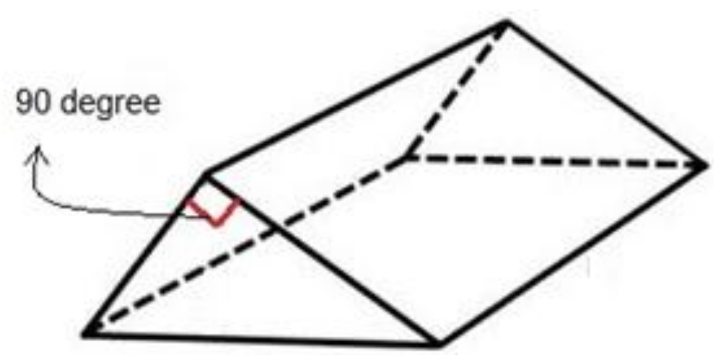


Figure 4.1: General form assumption

If the angle of the Triangular prism more than 90°,this is will cause a problem for the large book, such as damage in the book spine, and the small book will have a problem to sit on the saddle and have a missed scanned parts of the pages ,on other side if the angle is less than 90° that will increase the friction force between the saddle and system as discussed in chapter three in equations **3.1** and **3.2**and this will damage book with time of scanning process as shown in the **Figure4.2**.So the optimal choice of the angle is to be equal 90°.

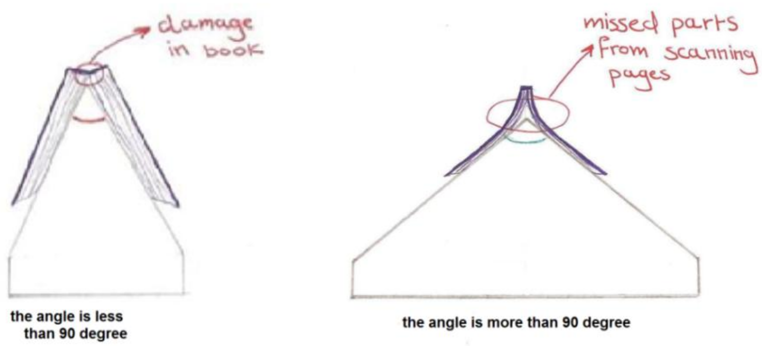


Figure 4.2: Triangular Prism angle for big and small books

The book should be placed turned down at Triangular Prism sides, to turn pages from starting side to another side through vacuum and slot channel , this operation need to move the book with saddle, so we connect the saddle with a belt that moved by the stepper motor.

The scanner and vacuum places depend on operations speed, the system starts with scanning the pages that is directly face the saddle, then flipping pages to scan the next two pages, so the scanner should be before the vacuum place, the distance between scanner and vacuum should be small, to reduce time cycle for the project, and reduce the total length of project.

4.3.2 Saddle design

The purpose of using the Saddle is to slide the book on Triangular Prism, and to link the book with the saddle, to avoid book vibrations that may cause damage in scanning, so the saddle design should be suitable for book style, book movement and linking method.

Saddle dimension depend on the size of books that will be scanned, the default books have pages at size A4 (21 cm * 29.7 cm), saddle dimension should be more than book dimension, so the size of the saddle should be more thanA4 size, the saddle should have movable wide supports changed by lead screw to suit with different size of the books, as shown in **Figure 4.3**

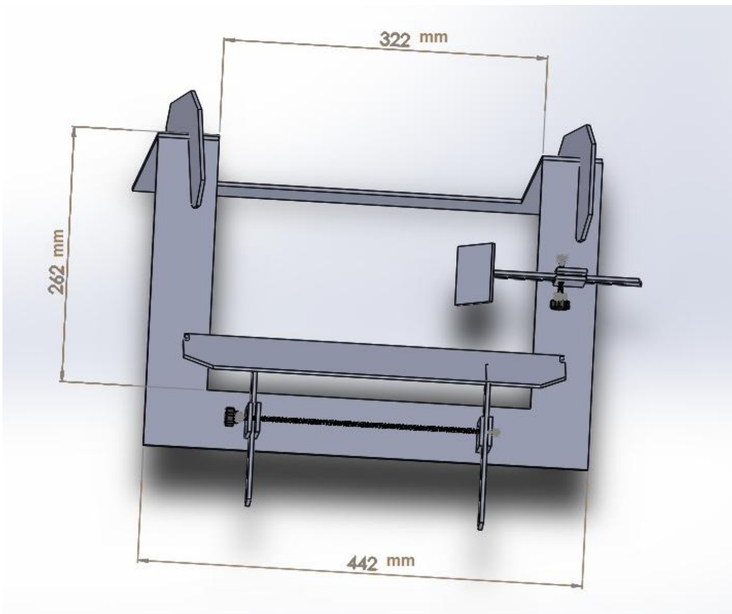


Figure 4.3: Saddle design

4.3.3 Vacuum design

In the previous chapter, the purpose of vacuum is to flip pages, so the vacuum should be suitable and variable for page size, it should consist two parts, one variable and the other fixed. The fixed one is a tube with slots that distribute the vacuum air, the variable one is a Mylar film sheet has an inclined line with holes, and we rotate it around the fixed tube, so this allows to change air distribution, depending on book size, as shown in **Figure 4.4.**

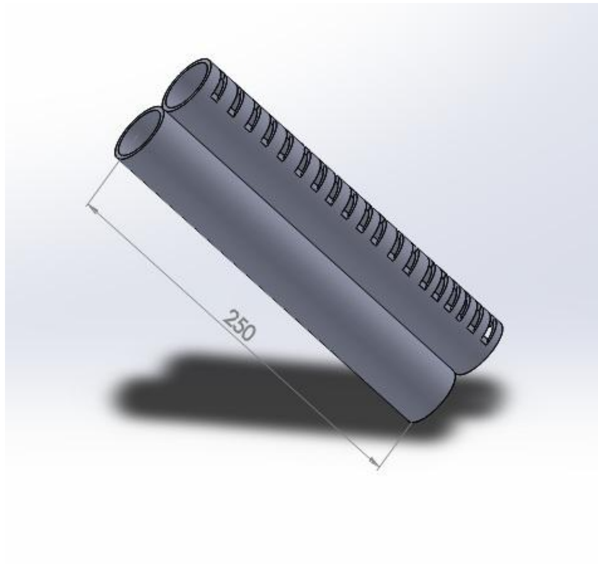


Figure 4.4 Vacuum tube

The slot channel used to pass the page from triangular prism's side to the other side, flipping pages starts in the going stage, and the page still in the slot channel as shown in **figure 4.5** by blue line, in the reverse stage, the page should move into the other side of from triangular prism as shown in **figure 4.5** by green line, the slot channel should have some specification such as smooth and inclined edges for easy movement of the page.

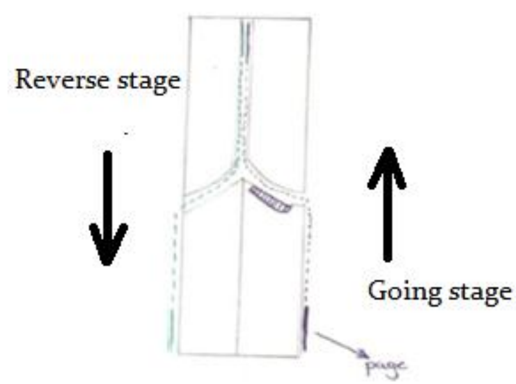


Figure 4.5: Slot channel design

4.3.4 Scanner location:

The two scanners were placed in the first part of the system before the vacuum channel, the two scanners are next to each other so they may scan at the same time, but are slightly offset in position so that the scanners can be mounted as close to the top edge of the system as possible as shown in **figure4.6**, and not interfere with each other. The scanners require a thickness of glass between the scanner and the page.

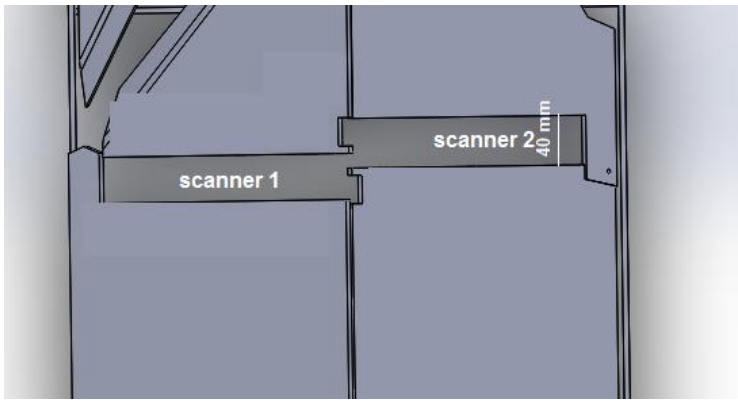


Figure 4.6: Scanners location

4.3.5Stepper Motors with belt

The purpose of using stepper motor is to move the saddle, as explained in the previous chapters, the saddle is attached to a timing belt which passes around and through the machine, the path of belt moving should take into account the saddle movement range, the motor location should be inside the frame, for safety and good appearance design, the belt path will be as a cycle, start from the saddle above the frame and continues to the motor below the frame, the shape of the belt's path as rectangular, so the path need a pulleys to change it direction as shown in **figure 4.7.**

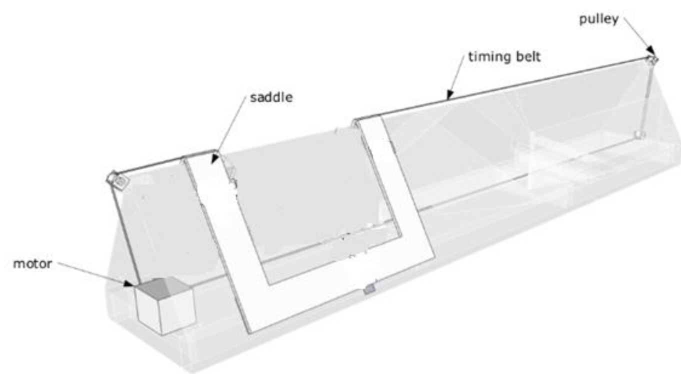


Figure 4.7: Timing belt's path

Pulleys selection depend on the belt's size that also depend on applied force from stepper motor, the stepper motor's applied torque explained in section 3.2.

4.4Material selection

Material selection depends on many factors such as cost, strength, fabrication, form, friction, the system can be designed from Stainless steel, Aluminum and Acrylic[1],Acrylic is the optimal choice with respect to the previous factors. By make a comparison between the three materials, Acrylic fabrication is more easy than the other, Acrylic gives the system light weight and good appearance and less noise, the friction between Acrylic and pages is less than the Stainless steel and Aluminum.

Acrylic specifications:

1) Tensile Strength: 69 MPa.
2) Modulus of elasticity: 2800 MPa.
3) Shear Strength: 62 MPa.
4) Forming Temperature: (170-190˚C).

To reduce the friction between the Saddle and the specific path in the frame, the saddle will has a special piece of Derlin[2] material, because it has low coefficient of friction.

[1] Appendix B
[2] Appendix B

4.5 Forces and Stress Analysis

4.5.1 Stress Analysis for side plate using Solidworks Simulation

Solidworks simulation provide many mechanical analysis to make sure that the design can hold the load and to make sure is that safe or not, in this project Solidworks will be analyze the side plate forces to determine maximum stress, displacement, strain and factor of safety.

Solidworks simulation window facilitate doing these analysis by sequential steps in study advisor, and in this advisor, each step has explanation for all probabilities in the design and analysis, the next steps will explain side plate stress analysis:

Step 1: select the desired part to do stress analysis, by opening the part file.

Step 2: select new study in simulations window.

Step 3: select the apply material, the material that used for manufacturing the frame is Acrylic, and it has 73 MPa tensile strength and 45 MPa yield strength.

Step 4: selection the fixed geometry on the part, the part has four pin position all should be assumed fixed, and the upper and lower sides also should be fixed.

Step 5: determine the magnitude and directions of the applied forces on the plate, which is the book weight (it assumed to be 3 KG maximum) and shear stress that produced from book movement.

Step 6: running the simulation and waiting the results.

Results

After finishing the simulations operations, the results will appear automatically with ability to show each test alone or together, the simulation analysis will give four analysis result which is:

1) Stress test :

As shown in **figure 4.8 –a**, the magnitude of stress in each point on the plate can be determined by using the color's contour, by notice it can be determined the maximum stress on the plate, it will be 0.17 MPa, and this value is too small, comparing to yield strength which is 45 MPa.

2) Displacement analysis :

The forces that affect on the plate will produce bending moment and displacement, this displacement may cause faults in system operations especially in scanning operations, **figure 4.8 –b** shows the displacements in each point on the plate, as noticed the maximum displacement will be 0.018 mm, this value is very small and it will not cause any faults in systems operations.

3) Strain analysis :

Engineering strain is expressed as the ratio of total deformation to the initial dimension of the material body in which the forces are being applied. The nominal strain (e) is expressed as the change in length ΔL per unit of the original length L of the line element, **figure 4.8-c** shows the strain values in the plate, the maximum value is approximately 5 ×[10]^(-5) and this value mean that the part has low deformation.

4) Factor of safety:

Factors of safety (FoS) is a term describing the load carrying capacity of a system beyond the expected or actual loads. Essentially, the factor of safety is how much stronger the system is than it usually needs to be for an intended load, the factor of safety values can be calculated by dividing the ultimate strength to ultimate load, **figure 4.8-d** shows the factor of safety values and the minimum values is approximately 100.

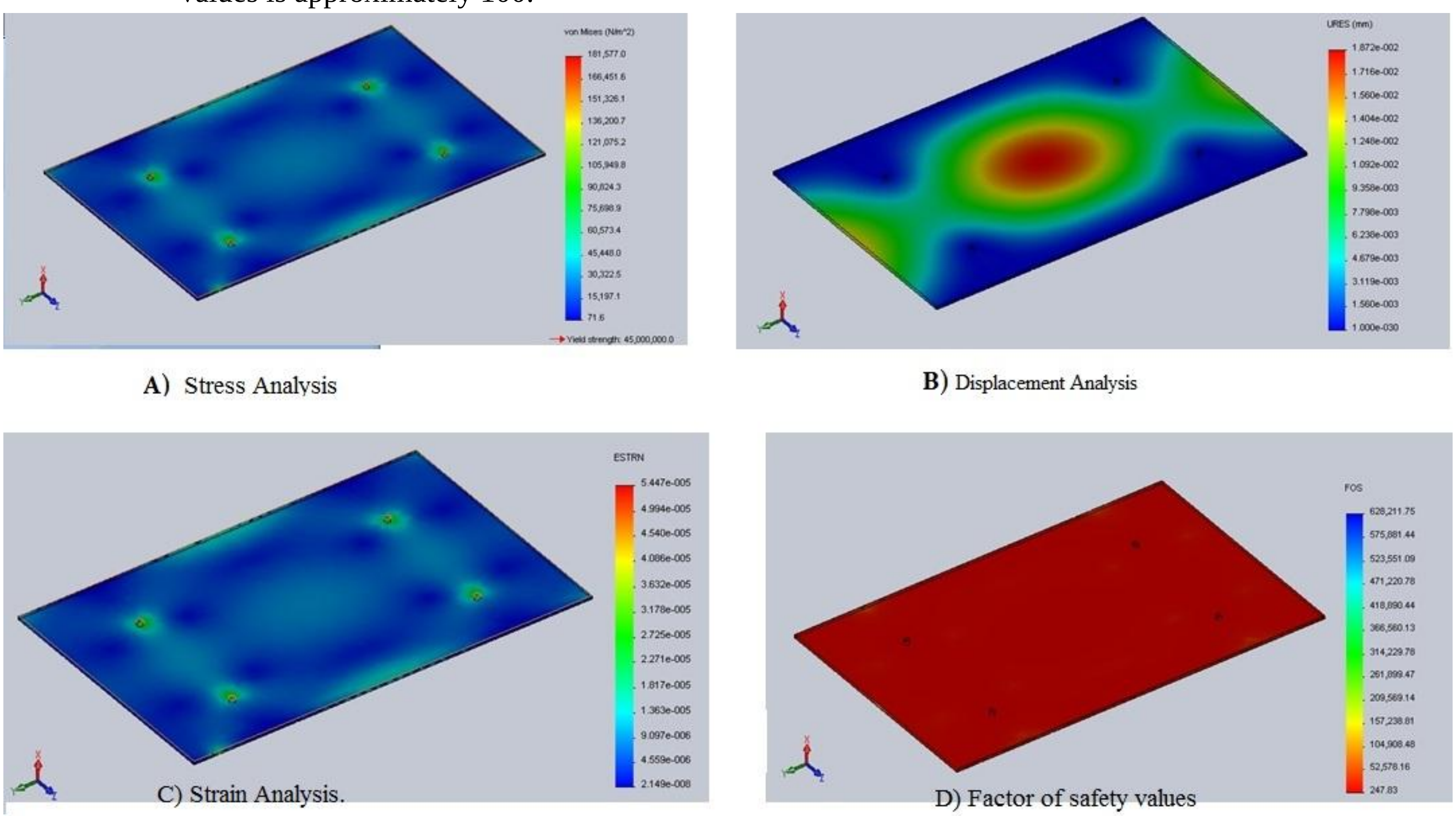


Figure4.–8: Simulation Results

The upper plates should be based on vertical supports, these supports help to resist force effect on the design and link the both sides together, and there is some considerations in support design:

1) Support's thickness, it depend on force analysis, and available material type in the market, mostly, it will be made from acrylic, its thickness 4 mm.
2) Supports shape: supports shape should considered to the purpose of using, in this case, its shape should be as a triangle, to link and support both sides of prism.
3) Supports dimensions: it should be appropriate to the plates size and its dimensions.
4) Other functions: the supports can be used to pass elements through it, for example some supports should have a space in its design to pass the flipped pages through it, and other supports should have a space in its design to put the vacuum and its tube.
5) Distance between supports: the different design of supports will cause a variability of force resistance for each supports, this problem can be solved by reducing the distance between vertical supports " less than 29 cm ", because the book weight will divided to the both supports, and the design will has strength ability of resisting the load.
6) The vertical supports need a horizontal supports to link all supports together, and to increase design strength.

Figure 4.9and **Figure 4.10**, shows the vertical and horizontal supports, its arrangement and its designs and the distance between supports, there is many differences between the design of each support, that depends on the page movement in the slot channel, how to set the vacuum tube on the supports and the arrangement of the parts inside the system.

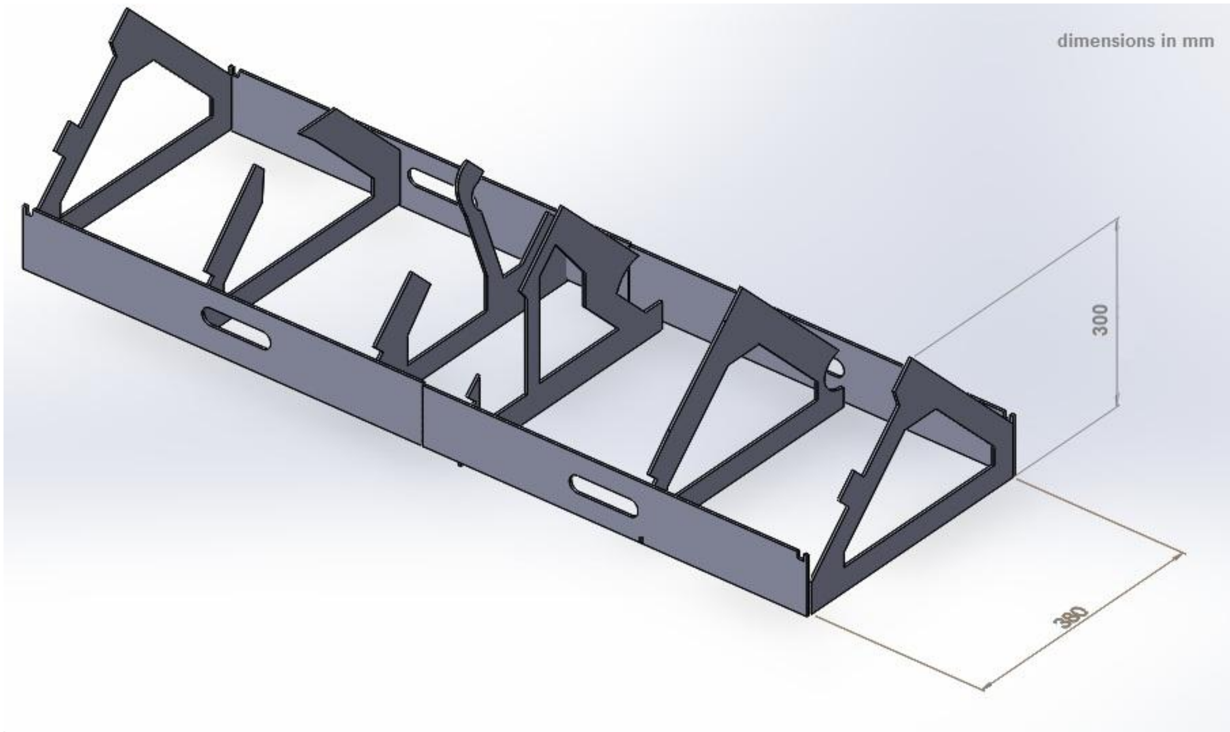


Figure 4.9: Supports design

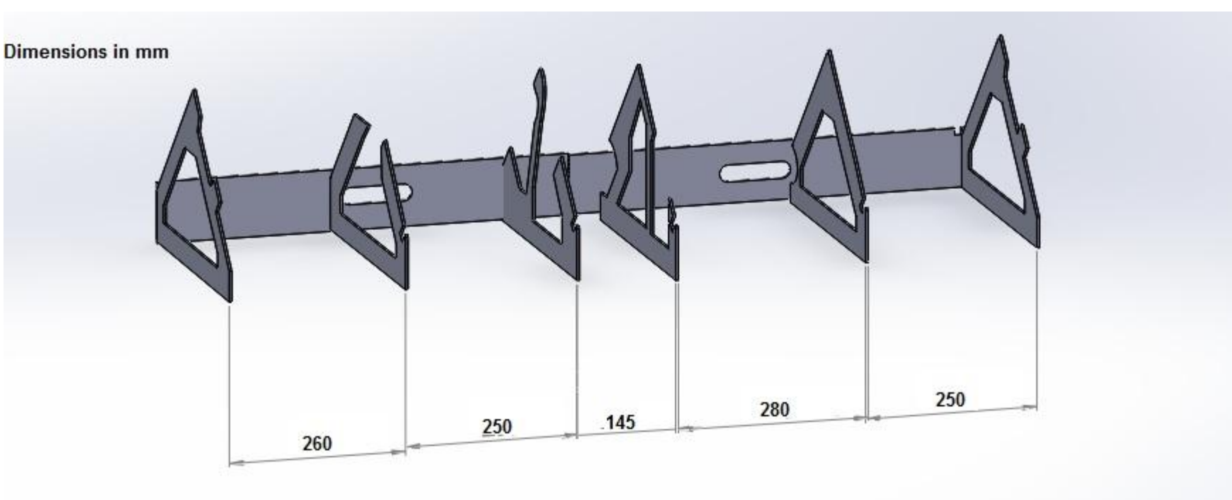


Figure 4.10 Distance between supports

4.6 Final Design

After design process for each element and identify each purpose of element, style and dimension, final design style is shown in **Figure 4.11.** Total length of project depend on many factors in the project, includes saddle length, scanner, desired speed of the system, slot channel, consumed time in each operation and additional distance for accelerate the motor's speed (between scanner and vacuum).Depend on these factors the total length of the system is 120 cm**.**

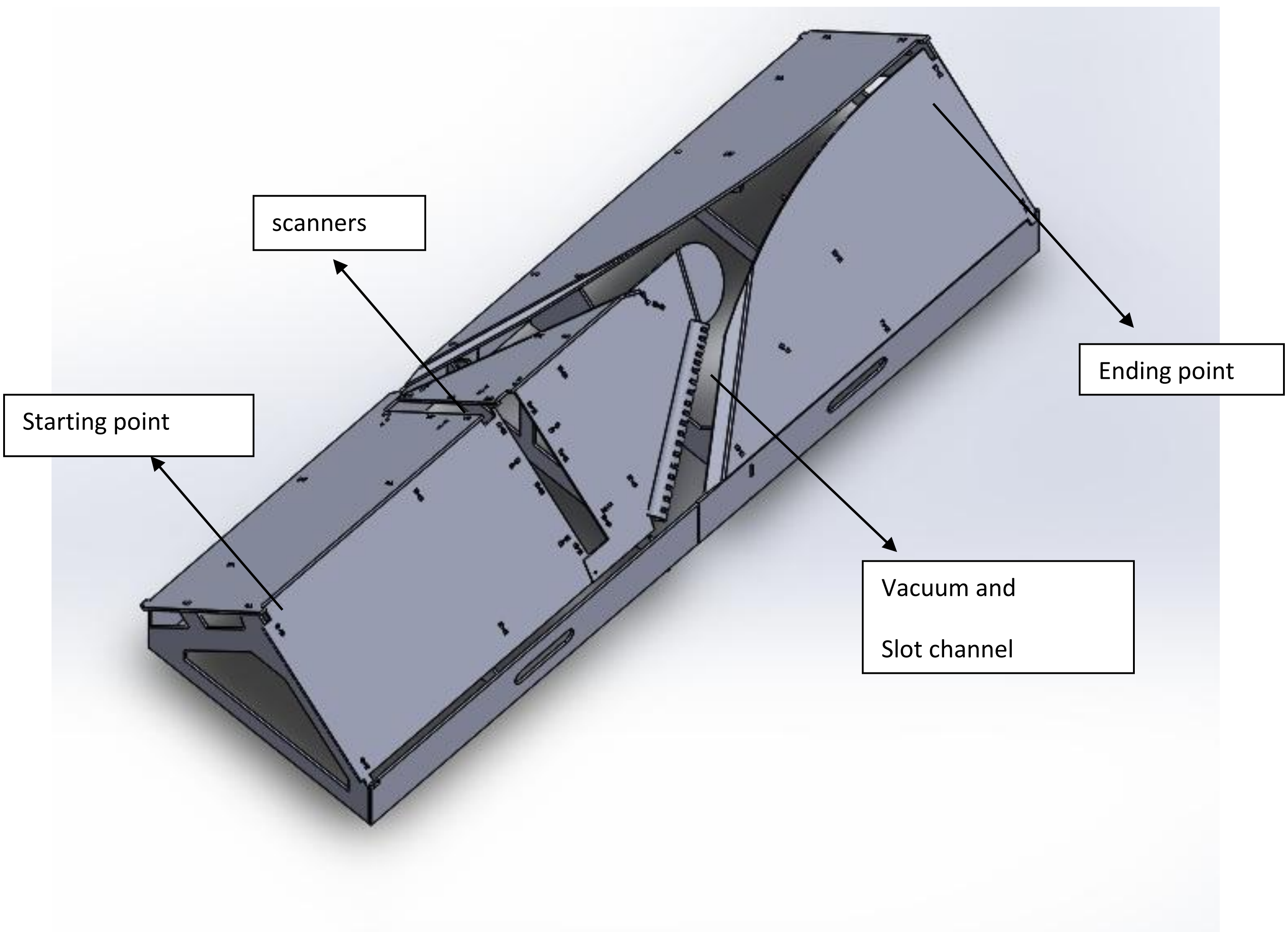


Figure4.11 Final Design

4.6.1 Main Parts of the System

In the following the main parts of the system:

1) Infeed Front Side: The infeed front side panel houses a scanning bar (scanner) and the vacuum tube. This Panel assembly is shown in **Figure 4.12**. Care was taken to mount the scanner bars as far into the corner as possible so that most of the books gutter may be scanned. The top leading edge of the infeed front side panel is designed to help the system to turn the page.

.

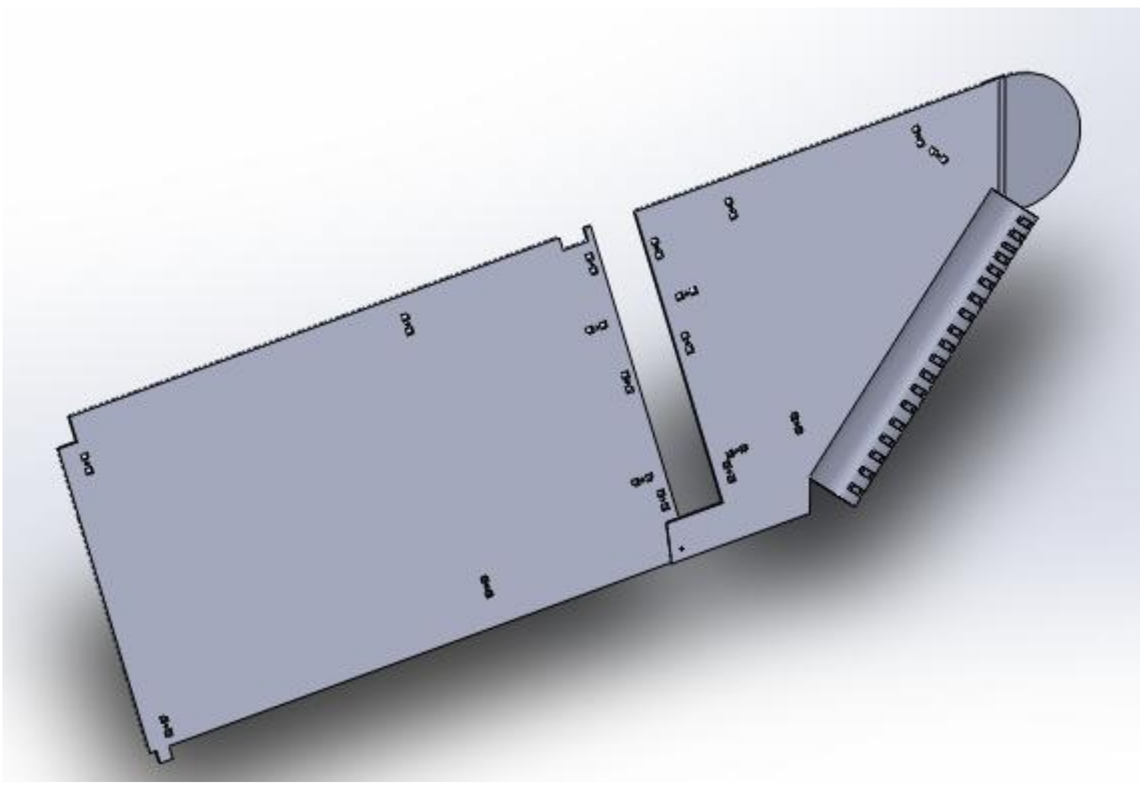


Figure 4.12 Infeed Front Side

2) Infeed Back Side**:** The infeed back side panel as shown in **Figure 4.13** is designed to have a bend curve part to allow the page to enter the channel and it has a place to put the infrared sensor.

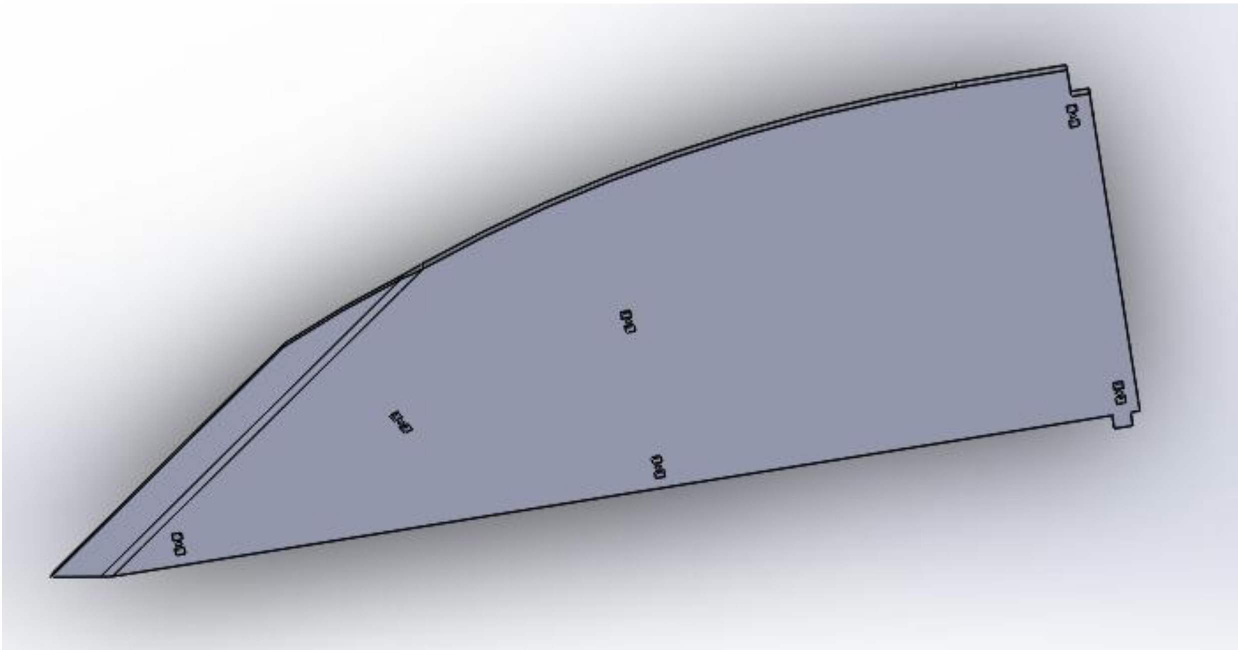


Figure 4.13 Infeed Back Side

3) Outfeed Front Side**:** The outfeed front side panel houses the second scanning bar and has geometry to help the page exit the channel. The scanning bar is offset from the infeed in order to push it higher so that more of the books gutter can be scanned. The bend is necessary to help the turned page to return to the path as shown in **Figure 4.14**.

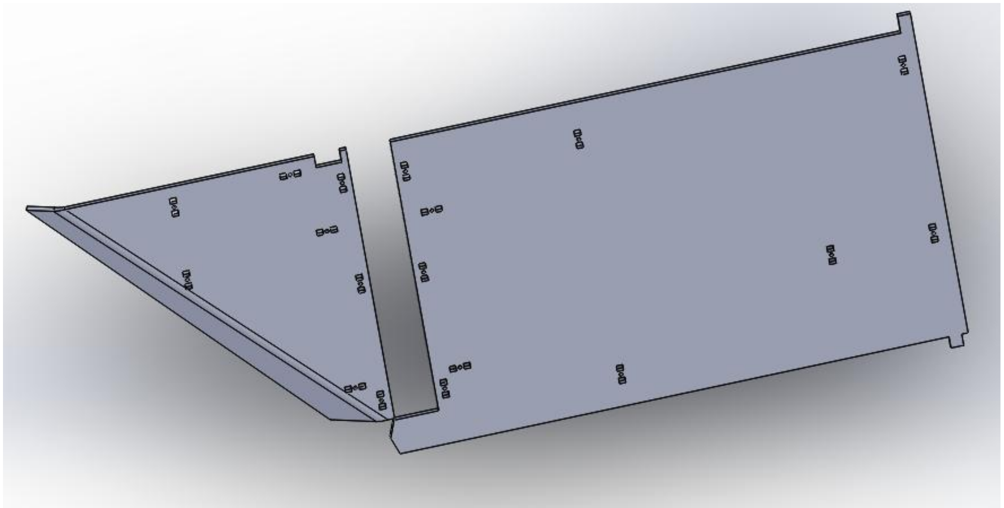


Figure 4.14:Outfeed Front Side

4) Outfeed Back Side: The outfeed back side panel is designed to have a bend to allow the page exit from the channel to outer surface of the system as shown in **Figure 4.15**

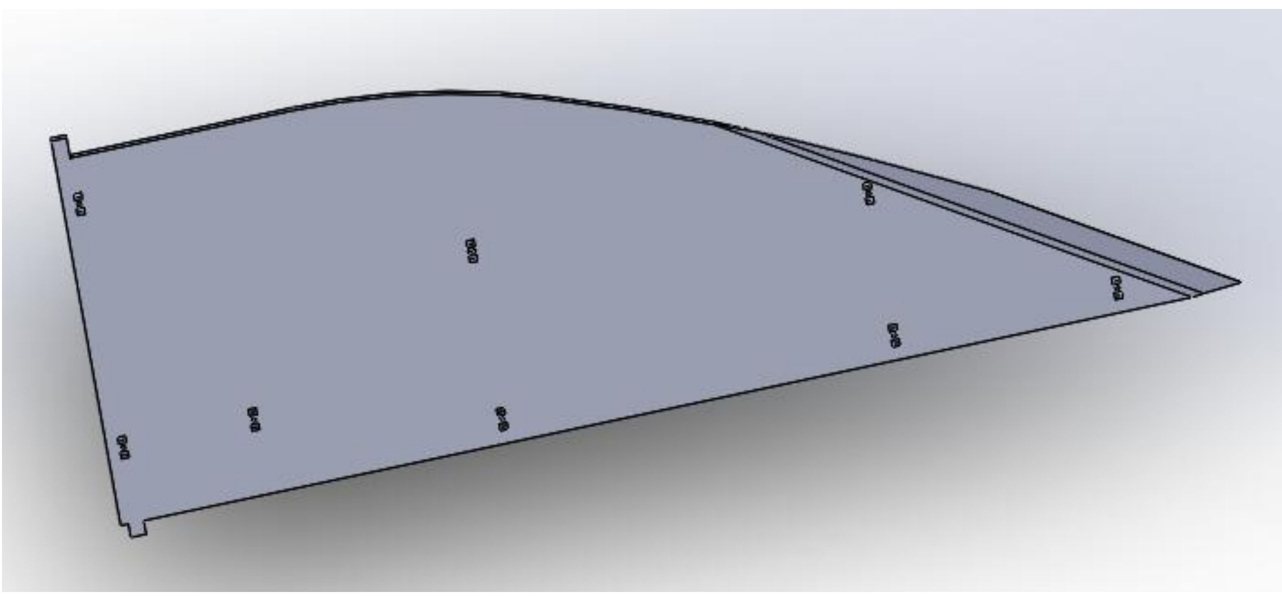


Figure 4.15 Outfeed Back Side

4.6.2 Assembly of the final design:

Figure 4.16 shows the final design parts and how to assemble them.

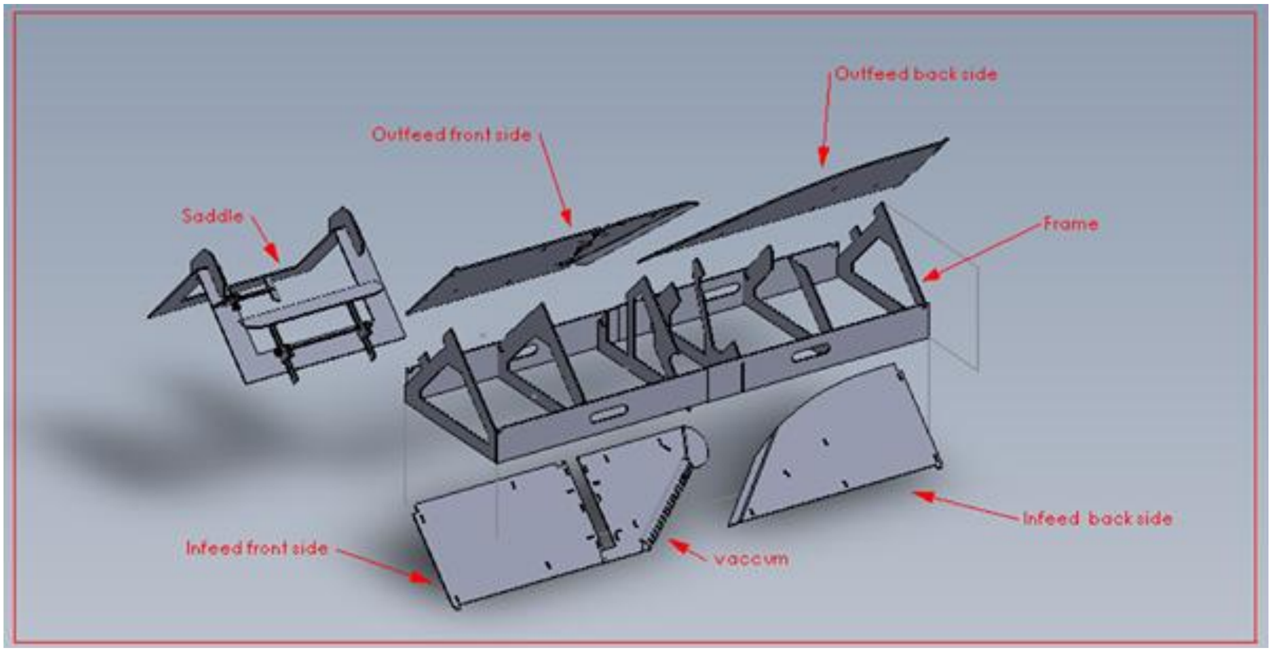


Figure 4.16 How to assembly the system

4.7 Electrical design

Previous chapters explained the purpose of using stepper motor, Arduino and Raspberry Pi , this section will explain the schematic diagram of connection between electrical components, and explain the relationship between electrical components .

Arduino used for controlling Stepper motor speed and direction, Nema 23 need a driver to connect to Arduino, the driver need +12V power supply for operate the stepper motor, and 4 pins used for receive the stepper motor movement signals, and the schematic diagram of the driver and Nema 23 is shown in **Figure 4.17.**

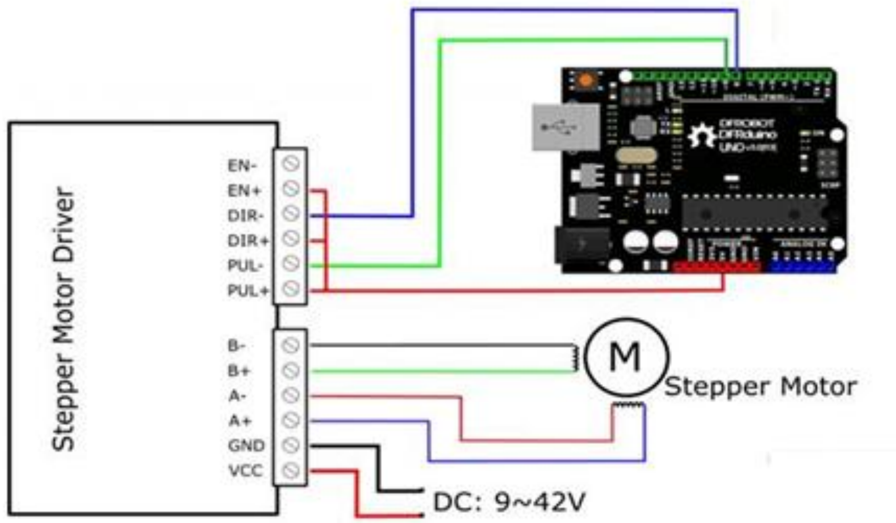


Figure 4.17: Schematic diagram for Arduino and stepper motor

As shown in **figure 4.18** Arduino and Raspberry pi is connecting to each other by using GPIO pins in Raspberry Pi and digital pin in Aurduino and an interface circuit was used to convert voltage from 5 volt to 3.3 volt.

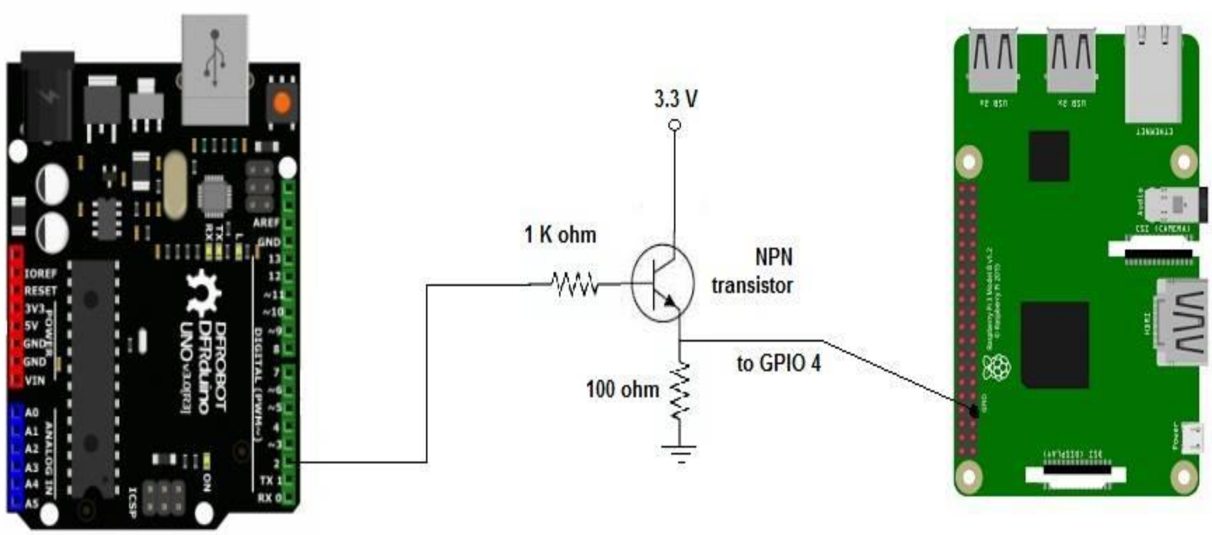


Figure 4.18: Arduino and Raspberry Pi connection

The scanner also connected to Raspberry Pi using USB, scanner running depends on the saddle movement location that can be provided from Arduino, and run the scanner by using Raspberry Pi program.

Raspberry Pi current limitation is approximately 2.5 A, the two scanners need1 A to run, each USB port and HDMI need 160 mA and the Raspberry PI need (800-1200) mA, so the Raspberry Pi cannot run the scanners because it will occur overload in Raspberry Pi current, so it will be used a USB hub with external power supply has 1.5 A minimum, **figure 4.19** shows USB hub.

Figure 4.19 :USB hub with external power supply

The vacuum device can be controlled by Arduino, when the saddle reach the vacuum device, Arduino get a command to vacuum device to run, this operation can run by comparing the stepper motor position with the vacuum device position, if this comparison occur, Arduino give a digital output (5Volt) on pin 8 to transistor and run the vacuum device, as shown in **Figure 4.20**.

Figure 4.20: Arduino and vacuum device connection

The limit switch and infrared sensor are used as feedback sensors, so its signal will be input for Arduino, and processing its data for determination operations cases by using C++ code, limit switch will give digital signal that will determine starting saddle position, the output pin from this sensor willtied with pin 12 in Arduino as shown in **figure 4.21**

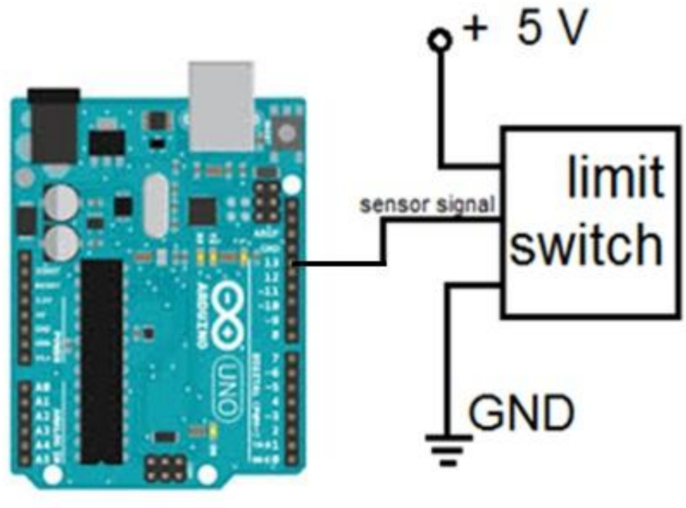


Figure 4.21: Limit switch and Arduino connection

The infrared sensor gives a digital signal, so it will be tied with pin 13 in Arduino, the digital signal give the system a high signal if there is a page in the slot channel and a low signal if there is no page in the channel, **figure 4.22** shows the sensor connection with Arduino.

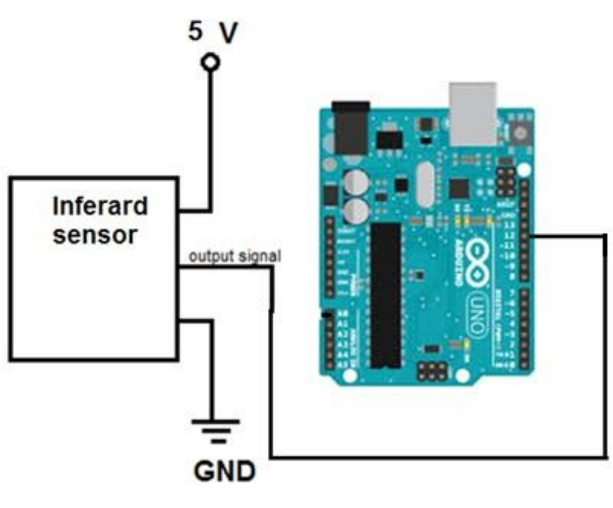


Figure 4.22:Infrared sensor and Arduino connection

5

Chapter Five

Fabrication and Assembly

5.1 Overview.

5.2 Structure Fabrication.

5.3 Structure Assembly.

5.4 Electronic parts installations.

5.1 Overview

The previous chapters discussed the hardware components, size effect on the structure design, and many factors that affect the structure. This Chapter discuss fabrication and installation of all components and parts.

5.2 Structure Fabrication

Depending on material selection in section 4.4, the best way to form the acrylic is to use laser cutter (2D). The laser cutter has a limitation for working area and cutting depth. The available laser cutter in West bank has limitation for depth and is equal to 4 mm, so acrylic sheet with 4 mm depth or less would be suitable. This point was considered previously, before final structure design.

The laser cutter requires 2D drawings for the project design, and it should to considered maximum area for structure parts, the laser working space area is 1 m *0.5 m, so it divided the 2 dimension draw into 7 parts, each part has 50 cm * 70 cm, and arrange the parts in shape that can exploit the area as much as possible, figure 5.1 shows the laser cutter drawing sections.

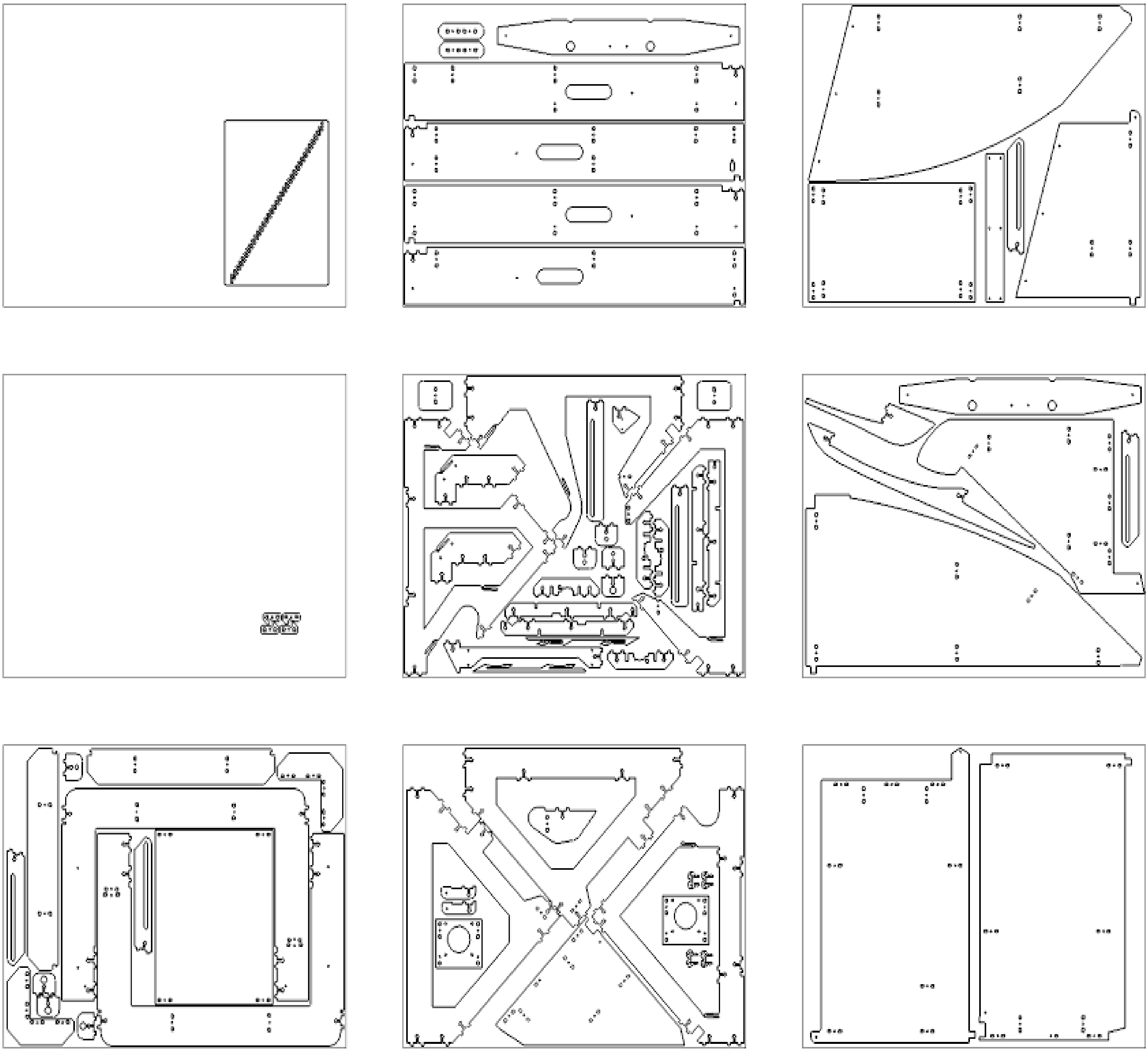


Figure 5.1 Laser cutting drawings

After laser cutting, the parts were checked for possible defects, **figure 5.2** shows the different parts after laser cutter fabrication.

Figure5.2 Parts after fabrication

The entrance and exit of slot channel should be suitable to page flipping, so it should have a bend in the end of side plates. This operation can be done by a plastic shaper of advertisement panels, using a heater. This is done carefully and professionally, because the shaper will saw small layer in acrylic and it will use the heater in bending operation, **figure 5.3** shows the bending operation.

Figure 5.3 Making the bend by the heater

The side panels in the structure are set with screws, where the type of screw should not block or oppose the book sliding. Those were chosen flat-head screws. The flat screws need drilling in the side panels at screw entrance, it can be done by using countersink. Figure 5.4 shows countersink in the works.

Figure 5.4 Countersink Working

The final step in fabrication is the vacuum tube fabrication. The vacuum tube as explained previously consists of two parts, the first one is fixed tube, and the other one is variable Mylar film sheet. This is used to change airflow and adapt it to the book page flipping. **Figure 5.5** shows the final fabrication and assembly for vacuum tube.

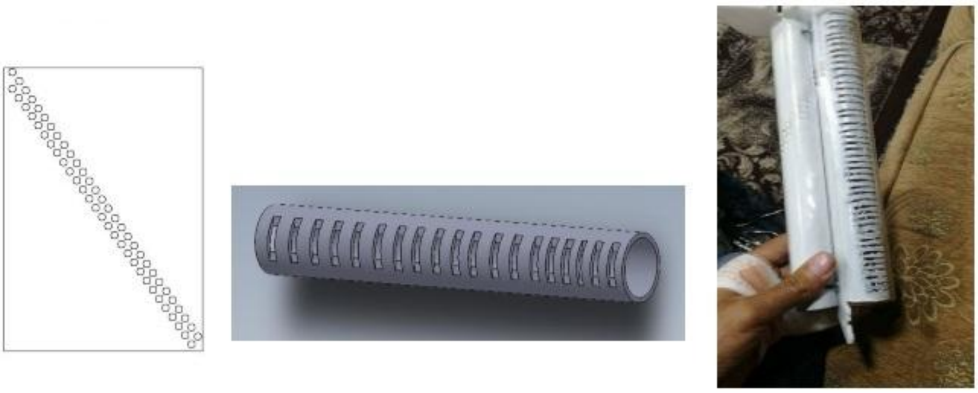


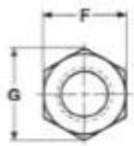
Figure 5.5 Vacuum Fabrication

5.3 Structure Assembly

There are two types of screws that are used in the structure assembly; First one is internal hex flat head cap screw, which is 16 mm length and 3 mm diameter, as shown in **Figure5.6 a.** The other type of screws is internal hex socket cap –head cap screw, which also has 16 mm length and 3 mm diameter, as shown in **Figure 5.6 b**. The nuts that are used for previous screws has 3 mm diameter and 5 mm width, as shown in **Figure 5.6 c.**

a)

b)



Nom Size or Basic Major Dia. of Thread		F			G		H		
		Width Across Flats			Width Across Corners		Length		
		Basic	Max	Min	Max	Min	Basic	Max	Min
1/4	0.2500	3/8	0.375	0.365	0.433	0.416	3/4	0.76	0.74
5/16	0.3125	1/2	0.500	0.489	0.577	0.557	15/16	0.95	0.93
3/8	0.3750	9/16	0.562	0.551	0.650	0.628	1-1/8	1.13	1.11
1/2	0.5000	3/4	0.750	0.736	0.866	0.839	1-1/2	1.51	1.49
5/8	0.6250	15/16	0.938	0.922	1.083	1.051	1-7/8	1.89	1.86
3/4	0.7500	1-1/8	1.125	1.088	1.299	1.240	2-1/4	2.27	2.22
1	1.0000	1-1/2/td>	1.500	1.450	1.732	1.653	3	3.03	2.97

c)

Figure 5.6: Screws of the system

After preparing all required screws and nuts, they need be fixed to assemble all parts together. Assembly would take horizontal and vertical supports. **Figure 5.7** shows the described assembly.

Figure 5.7 Assembly of the supports

The Fastener system, shown in **Figure 5.8**, facilitates assembly, by setting all supports together and then linking them using the screws and nuts.

Figure 5.8 Fastener System

After supporting the assembly, it should be assembled the side panels, it's assembly does not differs of the supports assembly, side panels should sets firstly using fastener system and then linking (joining) it by using screws and nuts, **figure 5.9**, shows the side panels assembly.

a) Supports

b) The final assembly of the panels and supports

Figure 5.9: The side panels assembly.

The saddle has same assembly operation, as shown in **figure 5.10**

Figure 5.10 Saddle Assembly

5.4 Electronic Parts Installations

First, set the stepper motor and the driver, **figure 5.11** shows stepper motor installation.

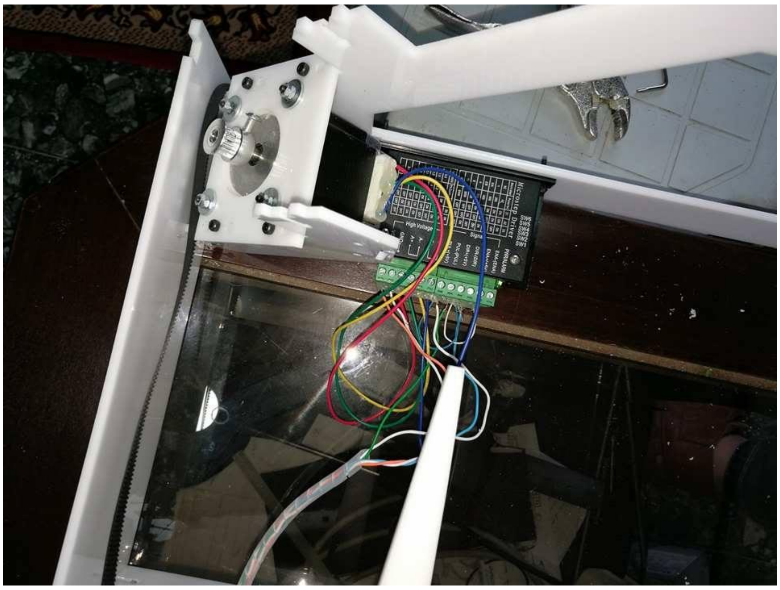


Figure 5.11 Steeper motor and driver location

Next step is to set the vacuum cleaner in the structure, it can be exploit the space in the structure to set the vacuum cleaner, **figure 5.12** shows the vacuum installation.

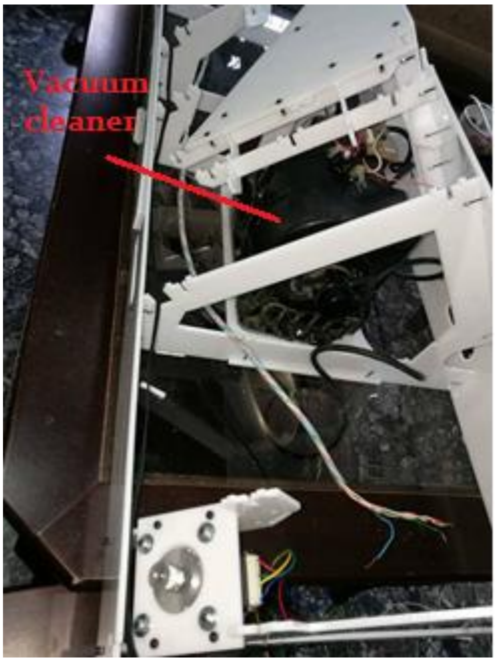


Figure 5.12 Vacuum cleaner location

Scanners installation need disassembling the scanner first then set each on the structure. The Cano Scan Lide 120 supports running without the sensors and motor is set in original structure of the scanner. The scanners and their drivers are shown in **figure 5.13.**

Figure 5.13 Scanner location

Then setting the switches, emergency switch and limit switch, as showing in **figure 5.14**

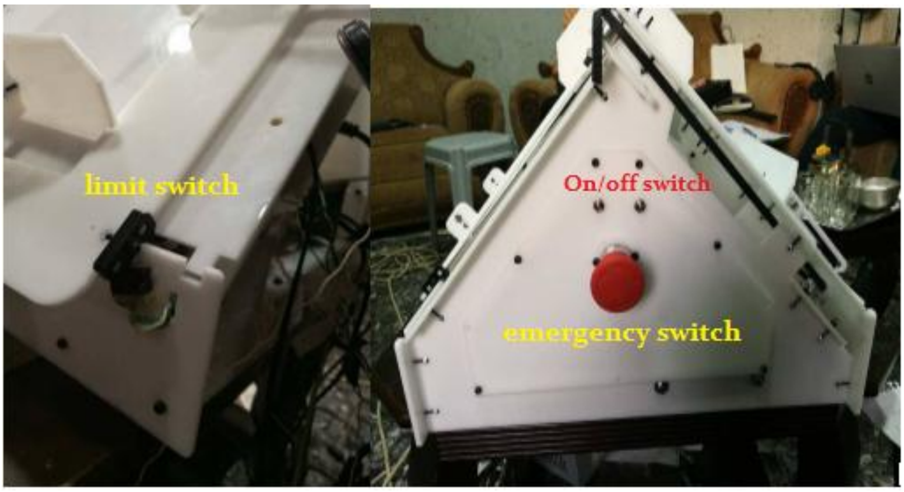


Figure 5.14: Switches, emergency and limit switch location

Then, setting Arduino, Raspberry Pi, USB hub and the power supply, and connect all the required wiring for all systems, as shown in **figure 5.15**

Figure 5.15 Control Unit

After finishing all assembly operation, programming Arduino and Raspberry pi would be the next phase for starting, operating and controlling the system along with necessary troubleshooting as would be explained in the following Chapter 6**.**

6

Chapter Six

Interface between Subsystems

6.1 Overview.

6.2 Properties of the Subsystems.

6.3 Determination the Systems Operators.

6.4 Arduino Code.

6.5 Raspberry Pi Programming.

6.1 Overview

This chapter will discuss the automated linear book scanner subsystems, its working properties, and how to interface all subsystems together. In addition the operational tests and troubleshooting; necessary for integration of the subsystems would be accomplished in order to achieve the objectives of this project.

6.2 Properties of the subsystems

This section will determine the subsystems and working properties and their integration together.

These subsystems are classified into:

1) The mechanical structure.
2) Scanning system.
3) Vacuum system.
4) Motion system
5) Feedback and control system.

6.2.1 The Mechanical Structure

The mechanical structure can be considered as a mechanism, because it has a similar purpose objective, which is to move the load; the book, so assume that the structure is a subsystem.

The previous chapters explained the factors and reasons of structure design. Such structure design depends on many factors, such as:

- The motion of the book.
- The arrangement of main processes.
- The size of component that used in the system.
- Material selection and forces analysis.

Any change in the other subsystems, will affect the structure design. **Figure 6.1** shows the final structure design.

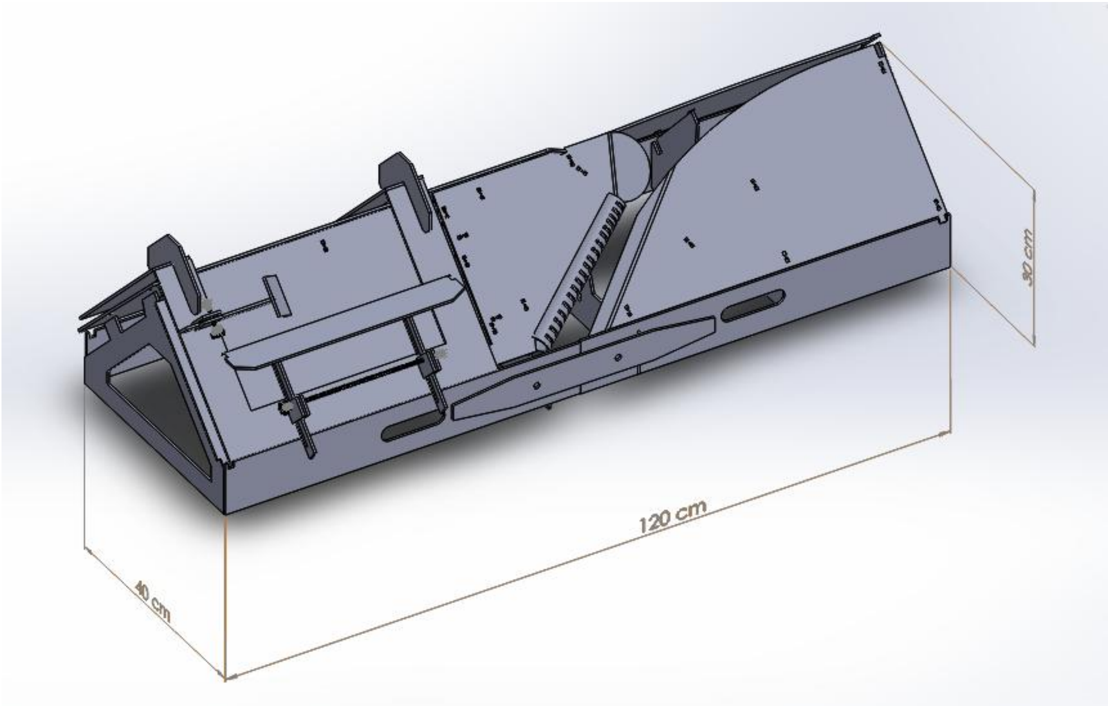


Figure 6.1 Final design of the system

6.2.2 Scanning System

Previous chapters, especially chapter 3, identify the scanner specifications. It is important to identify the scanner variables, to achieve most adequate performance with the other subsystems.

The most important variables factors for the scanner are:

1) Scanning speed.
2) Period of time for saving the scanned picture.
3) Scanning resolution.
4) Picture extension.

Scanning speed depends on scanning resolution, at resolution 300 dpi, it takes 13 seconds, but at maximum resolution (2400 dpi) it takes 60 second, **Table 6.2**, shows the scanning time for each resolution.

Table 6.1 Relationship between resolution and time.

Resolution (dpi)	Time (seconds)
300	14
600	30
1200	45
2400	60

Considering the requirements of the project, the optimal choice for time is at resolution 300 dpi, where it takes 14 second. It can achieve the speed that needed for the scanning system, it is approximately 300 page per hour, it is important to calculate the speed that book should move when it is under scanning and the speed can be calculated by dividing scanner glass length to the time of scanner running.

$$scanner\ speed = \frac{scanner\ glass\ length}{running\ time} = \frac{30.5\ cm}{14\ second}$$
$$= 2.178\ cm/second$$

There is another factor that should be considered for first scanning image. The scanner needs to test if there light on scanner sensors or not, which takes approximately 2 seconds. After scanning the first page, the scanner will run without testing the light, and the total scanning time will be 13 seconds.

By using python programming, running the scanners can be controlled, as well as the choice of the resolution that meets the requirements. The image extension can be identified in python code; there are many extensions available for Cano Lide Scan120, such as JPEJ, PNG and PDF, the perfect extension that meets with time, resolution and size requirements is JPEG, it has good resolution, low size and low time to save the image, approximately 2 second.

6.2.3 Vacuum System

There are three things that should be considered for the vacuum system:

1) The starting point when the vacuum is run
2) Period of running time.
3) Magnitude of force that the vacuum provides.

Returning to the vacuum specification, it can be noted that the vacuum has AC power supply, so it needs a relay to attach the vacuum cleaner with Arduino signal. Arduino code can run the vacuum at the required position of book movement. The period of running time depends on page length and the magnitude of force that needs to flip the page inside the slot channel. Notably, the Hemilton vacuum cleaner can change the flow rate of the drawn air. In addition by using the variable Mylar film sheet on the vacuum tube, the system can further control the change in the air flow and the length of drawn page area.

6.2.4 Motion System.

The previous chapters explained how much of torque is needed to move the saddle. As a result, two Nema 23 motors were selected to move the saddle and the book. The direction of motion and how to change belt movement using double pulleys on the corners were discussed.

Using stepper motor drivers, these will control the Nema 23 stepper motor easily using stepper driver library in Arduino. Such library can lead to controlling the stepper motor through two pins; the first pin is direction pin 0 volt give counter clock wise rotation, 5 volt give clock wise rotation. The other pin is pulses pin, it determine rotary angle and rotary speed.

The stepper motor has a driven pulley, which is attached between rotation of stepper motor, and translation motion of the belt that will move the saddle. The magnitude of translation displacement depends on the outer diameter of driven pulley; the outer diameter is approximately 0.25 inch (0.6125 cm), the circumference of the pulley is approximately 4 cm. This is verified experimentally by programming the stepper motor to rotate one revolution and measure the translation displacement on the structure.

It is very important to identify the displacements and speeds that are needed in each process. Such identification depends on the scanning and vacuum system. Motion parameters should work in harmony with the subsystems.

Each stepper motor has a driver which can control the stepper motor. The driver provides a multiple micro step property. The micro step specifies the magnitude of rotation displacement and rotation speed.

Using stepper driver library, this facilitates controlling the stepper motor. The library has fast working commands for choosing the rotation angle and magnitude of speed. The micro step value affects the angle and speed. The actual value of rotation angle and speed are divided into micro step values. Equation 6.1 and 6.2, explains the relationship between micro step and actual value of rotation angle and speed.

$$\text{Actual speed} = \frac{\text{speed value at arduino code}}{\text{micro step value}} \qquad \textbf{(6.1)}$$

$$\text{Actual rotation angle} = \frac{\text{rotation angle at arduino code}}{\text{micro step value}} \qquad \textbf{(6.2)}$$

The driver provides 6 values. These are 1, 2, 4, 8, 16 and 32.When the micro step increases, the actual value of rotation angle and speed will decrease. Table 6.2, explains the actual values at multiple micro step values.

Table 6.2 Micro steps values

Arduino code values		Micro step	Actual values	
Angle(degree)	Speed(rpm)		Angle(degree)	Speed(rpm)
360	100	1	360	100
360	100	2	180	50
360	100	4	90	25
360	100	8	45	12.5
360	100	16	22.5	6.25
360	100	32	11.25	3.125

By converting the scanner speed to revolution per minutes, 2.178cm/sec is equivalent to 32.67 rpm.

The benefits of choosing large micro step, is that it gives a precise displacement and precise low rotation speeds; at micro step 1, it cannot rotate the motor at low speeds (lower than 50 rpm), and it cannot give precise displacement, because the displacement for each step is big. However when the micro step value equal 32, it mean that the displacement for each step will be small, but more precisely

So the optimal choice for the system micro step is 4 because it provides the system with the required range of speeds.

Another property for the stepper motor driver, that it can specify the amount of current in the stepper. The current depends on the mechanical load. It can be set on the maximum value (4 A), with the presence of a power supply which can provide the current that is required for the stepper motor.

6.2.5 Feedback and Control System

Any mechatronics system should have a feedback for controlling systems. In our project, it required 3 feedback sensors. These were used to check whether or not the system works efficiently, and to correct for any consequent problem that may arise.

The sensors that are used for feedback and controlling are:

1) Limit switch : it is used to set the saddle at the starting point as shown in **Figure 6.2**
2) Infrared optical sensor: it is used to check if the vacuum flips the page or not.
3) Start and stop switches : they are used to start the system and stop (if needed)

Stepper motors do not need a feedback for displacement and speed, but need to identify where and when movement should start. At the point of starting the system, it should move back to the starting point to start scanning pages (homing the system), it can stop the saddle at starting point using limit switch, which is mounted at the end of horizontal support.

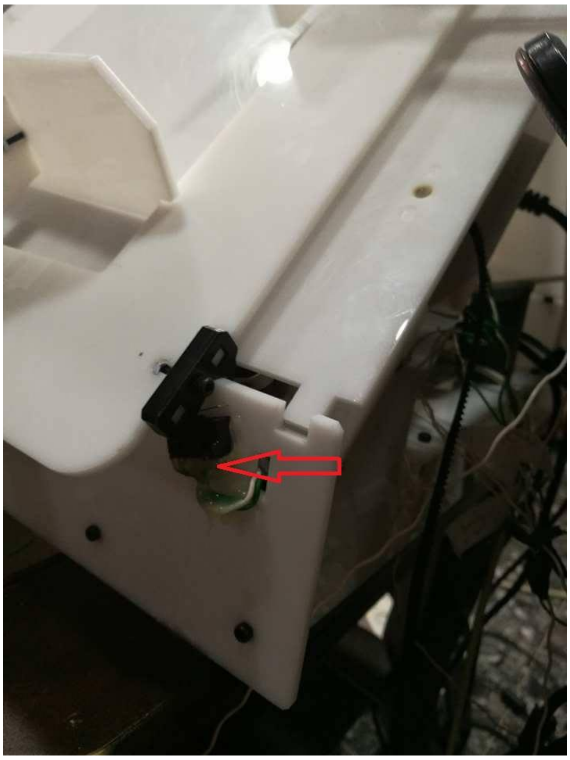


Figure 6.2: Limit switch of the system

The second feedback sensor is an infrared optical sensor as shown in **Figure 6.3**, it used to check if the vacuum flip the page or not, if the vacuum does not flip the page, the infrared optical sensor should send a signal to Arduino, and the Arduino should move back the saddle and run again the vacuum and check again if vacuum flip the page or not, after flipping the page, the saddle will go on to end of the system's structure and move back to starting point to start a new cycle and a new scanning process.

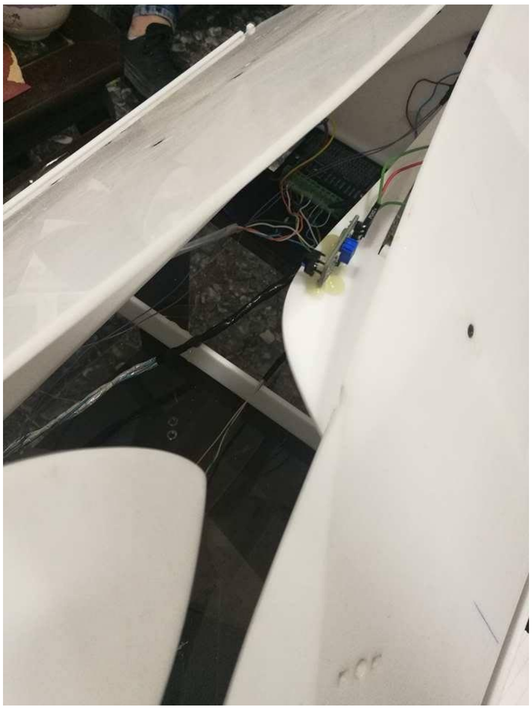


Figure 6.3 Infrared Sensor of the system

In certain some situations, the system should be stopped urgently. For example, if there is any probability to damage the book, there should be an emergency switch to cut off the power in the motors. In addition to use the start and stop switch to stop Arduino for sending signal to the scanners and vacuum, and the basic three systems will stop immediately. **Figure 6.4** shows the switches and the emergency switch of the system.

Figure 6.4 Emergency and switches of the system

6.3 Determination the Variables of Subsystems.

After determination of subsystems properties and parameters, in this stage, it should be start to calibrate all subsystems parameters to run all subsystems together smoothly and to make an integration between them, as defined in the previous chapters, Arduino will control all other subsystems, because running the other subsystems depend on book position and its movement, and the motion system controlled by Arduino, so it can be assumed that Arduino is the reference (the master) of the other subsystems.

At first, it should be determined the distance and speeds for running the other subsystems, and determined when each system should start. It is preferred to make a time line for the system based on a structure dimension, it will facilitate to understand process arrangement, and writing the Arduino code.

The system works in loops, which means that it will scan page and flip it and move back, and repeat these process until reaching the required scanned pages, but there are things that should never repeat, such as homing process, it is just done at the first running of the system, this point should be considered in Arduino code, **Figure 6.6** shows the arrangement of process in the system.

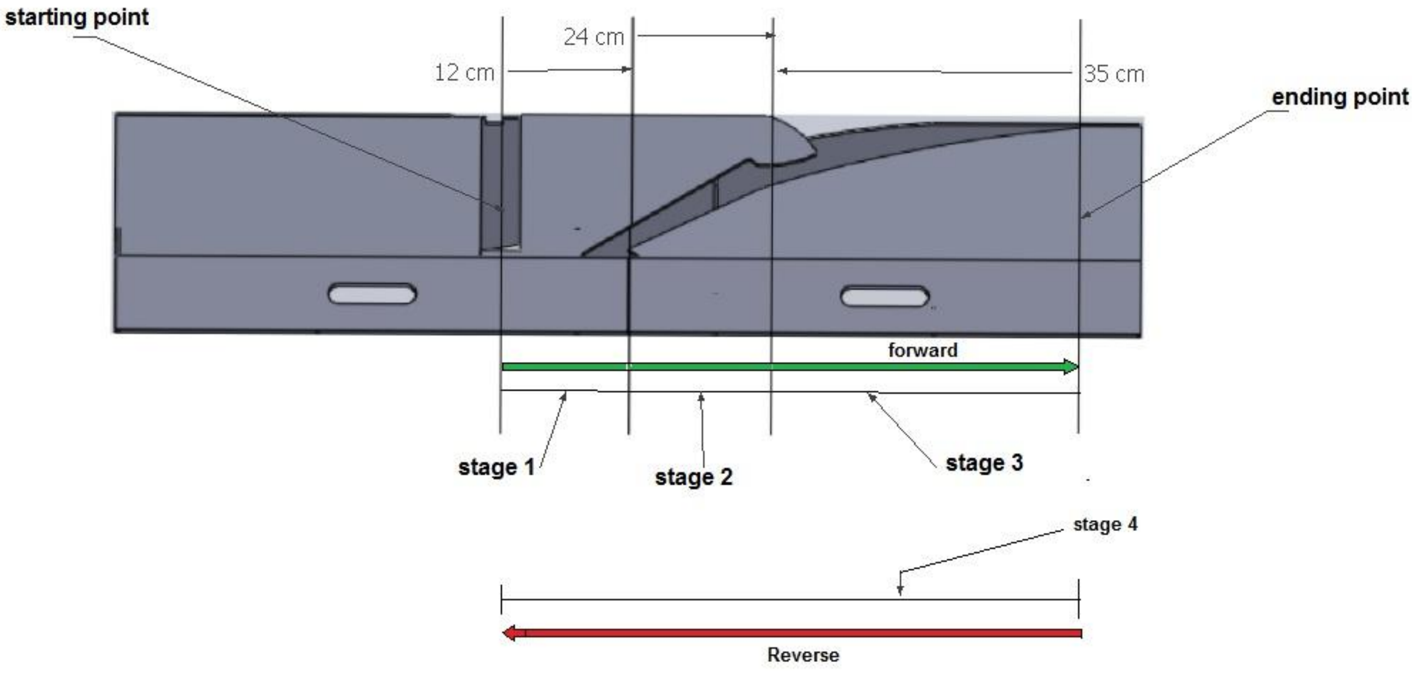


Figure 6.5: The sequences of system processes

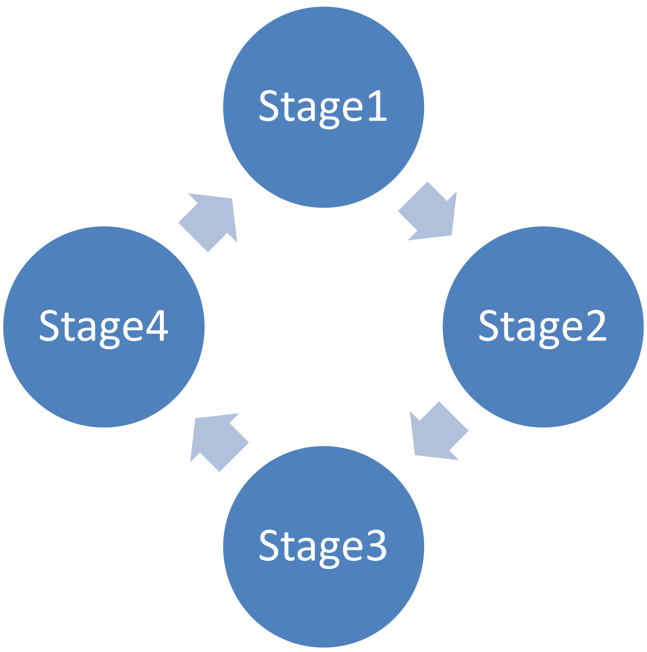


Figure 6.6: The stages of system processes

As shown in **figure 6.5** and **figure 6.6**, the time of working cycle divided to 4 stages, each stage has different parameters, it can be summarized the stages parameters as follows :

> Stage 1: at this stage the scanner should run, and saddle movement speed should be suitable to scanner speed, as calculated in section 6.2.The vacuum will not run at this stage.
>
> Stage 2: at this stage, the vacuum should run, the saddle at this stage will move as noted in the previous stage. The scanner would still be running.
>
> Stage 3: the vacuum and scanner will not operate at this stage, so it should be turned off, there is no reason to keep the speed same to the scanning speed, it should be increased the speed to decrease cycle time, to meet with desired cycle time in the project needs.
>
> Stage 4: in this stage, the saddle should move in the reverse direction, while keeping the high speed, also there is no need to run the scanners and vacuum.

After determining the stage parameters, it is time to write Arduino code which take into consideration these parameters with other subsystems properties**. Figure 6.7** shows the subsystem interface.

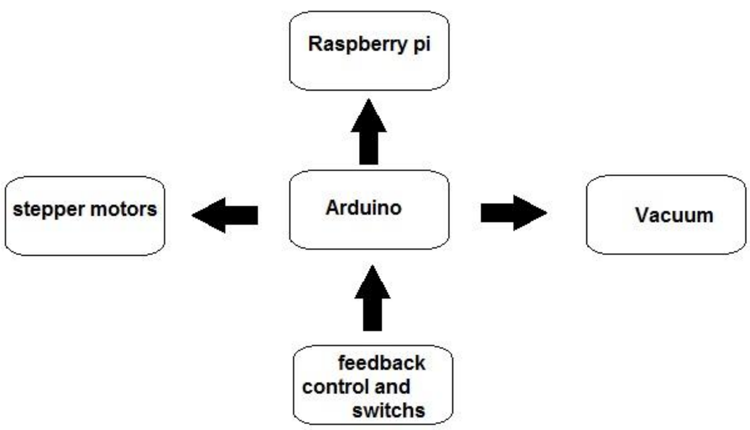


Figure 6.7 Subsystems Interface

6.4 Arduino Code

After determination subsystems operators and determination the time line for stages on the structure, it is easy to write the code, and represent all cases in Mikro C programming.

The Arduino code starts by defining the libraries used in the code, and definition of stepper motor driver pins, the following code shows the definitions.

```
#include <Arduino.h>
#include "BasicStepperDriver.h"
#define MOTOR_STEPS 200
#define DIR 8
#define STEP 9
BasicStepperDriver stepper(MOTOR_STEPS, DIR, STEP);
```

Then, it should define the output and input pins for the system, as well as other variables that are used in the code. The following code shows pins definitions.

```
const int scanner =  2;
const int vacuum =  4;
const int infrared = 12;
int inf = 0;
const int limitswitch = 13;
int ls = 0;
const int startandstop = 7;
int ss = 0;
```

The next step in the code, defines the pins setup. This operation is done for one time at the start of the code. It can set system homing in void setup, because it is also done for one time, the following code shows these options.

```
void setup()
{
  Serial.begin(9600);
stepper.setRPM(180);
pinMode(scanner, OUTPUT);
pinMode(vacuum, OUTPUT);
pinMode(startandstop, INPUT);
pinMode(limitswitch, INPUT);
pinMode(infrared, INPUT);
digitalWrite(vacuum, HIGH);
digitalWrite(scanner, HIGH);
```

The speed of reverse movement should be a slower movement, so it is set on 45 rpm actually, in Arduino code, it is 180 rpm because the microstep equal 4 on the driver, the magnitude of reverse movement probably be maximum length of the structure, so it should be supposed that the reverse movement equal 73 cm, the stepper programmed to move 2 mm and check if the saddle reach the starting point using the limit switch, the vacuum cleaner and scanners programmed to run on if the Arduino signal was low, the following code shows the reverse movement programming.

```
ss = digitalRead(startandstop);
while(ss == LOW )
        {
         ss = digitalRead(startandstop);

          }

for(int x=0 ; x< 365 ; x++)
{
 ls = digitalRead(limitswitch);

  if (ls == HIGH)
{
 x = 366;
 break;
  }
      ss = digitalRead(startandstop);

       while(ss == LOW )
        {
         ss = digitalRead(startandstop);

          }
  stepper.rotate(-72);

  }
```

Using for loop it can represent this operation, by move 2 mm by every time in loop iteration, the total distance equal 73 cm, so the loop iteration should iterate 730 mm /2 mm and it equal 365 once for iteration loop, the stepper should rotate 18 degree to move the saddle 2 mm on the structure, in the code it is set to be 72 degree, because the micro step set on 4, so the actual rotation will be rotation angle in the code divided to micro step depending on equation 6.2, when the saddle reach the limit switch, the Arduino will stop the reverse movement by set x=366 and break the loop, and the saddle will be at the starting point as required.

It can be used this step to start scanners for first time, because the scanner first scanning need more time, the Raspberry Pi programmed to ignore the first two pages, raspberry pi was programmed to start scanners if the Arduino signal equal low, after scanner running it should be wait 25 seconds to finish the scanning for first time of scanning, the following code shows first running of scanners.

```
digitalWrite(scanner, HIGH);
delay(2000) ;
 digitalWrite(scanner, LOW);
 delay(25000);
```

The next step in the code is to program the iteration movement, which is the 4 stages that explained in section 6.3, stage 1 is the scanning stage, so it should be set speed again, actually it is equal 32.67 rpm, it is calculated after finding movement distance for each revolution, and relate it with scanning speed in section 6.2.2, The total length for stage 1 equal 120 mm, by dividing 120 mm into 2 mm, this will lead to have 60 loop iteration to cover the full stage 1, the stage 1 should not start if the systems sets on stop at the switch, also it should to stop if the switch change from start to stop, it is used infinite loop when stop is set, if the switch change to start the stage will move on.

```
    ss = digitalRead(startandstop);

    if (ss == HIGH )
    {
    stepper.setRPM(130.7);
  for( int y=0 ; y< 60 ; y++)    //stage 1
  {
    digitalWrite(scanner, HIGH);
    digitalWrite(vacuum, HIGH);
    ss = digitalRead(startandstop);

    while(ss == LOW )
    {
      ss = digitalRead(startandstop);

      }

    stepper.rotate(72);



    }
```

The next stage is stage 2, vacuum stage, it should start the vacuum by changing pin signal to low (0 volt) and change scanner pin signal to high, to prevent scanners to scan again, this thing may occur problem in the system, the stage length equal 24 cm, so the for loop iteration should be 120 time, and should check start and stop switch, as shown in the following code.

```
for( int z=0 ; z< 120 ; z++) //stage 2
{
 digitalWrite(scanner, LOW);
  digitalWrite(vacuum, LOW);
  ss = digitalRead(startandstop);

  while(ss == LOW )
  {
    ss = digitalRead(startandstop);
     digitalWrite(vacuum, HIGH);
    }
    digitalWrite(vacuum, LOW);
  stepper.rotate(72);

  }
```

After this stage, it should be check infrared optical sensor signal, the sensor give 0 volt if there flipped page, if not, the saddle should move back and start vacuum again to make sure that pages was flipped, the following code shows infrared optical sensor programming.

```
  inf = digitalRead(infrared);
if (inf == HIGH)
{
stepper.rotate(-8640);
digitalWrite(vacuum, LOW);
stepper.rotate(8640);
digitalWrite(vacuum, HIGH);
}
```

The next stage is stage 3, in this stage it should stop the vacuum and scanners also, and it should check start and stop switch, the stage length equal 35 cm, and it is equal 350 time for iteration loop, and there is no determents for the speed in this stage, so it can increase the speed into 300 rpm, in the code, the speed set on 800 rpm because the micro step equal 4, the following code shows the stage programming.

```
stepper.setRPM(800);
for( int c=0 ; c< 175 ; c++) //stage 3
{
 digitalWrite(scanner, LOW);
  digitalWrite(vacuum, HIGH);
  ss = digitalRead(startandstop);

  while(ss == LOW )
  {
    ss = digitalRead(startandstop);

    }

  stepper.rotate(72);


  }
```

The final stage is stage 4, it is the reverse movement of saddle, the saddle should return to the starting point to start new cycle, the stage length equal 71 cm, it is equal 355 time for iteration loop, the speed still as the previous stage, to make sure that cycle time fit with needs, the total time for cycle equal 22 seconds and it is achieved the required need, the following code shows the stage programming.

```
for( int n=0 ; n< 355 ; n++) //stage 4
{
 digitalWrite(scanner, LOW);
  digitalWrite(vacuum, HIGH);
  ss = digitalRead(startandstop);

  while(ss == LOW )
  {
    ss = digitalRead(startandstop);

    }

  stepper.rotate(-72);

  }

delay(200);
```

6.5 Raspberry Pi Programming

Raspberry Pi is responsible to run the scanners. When the saddle reaches scanning point, exactly at starting of stage 1, Arduino automatically sends a signal to Raspberry Pi to turn on the scanners. After scanning a couple of pages and saving the images, Raspberry Pi waits for the next cycle to scan again and waits for the next signal from Arduino, after scanning. Raspberry will convert JPEG scanned images into PDF files, and merge them finally. Raspberry Pi can publish the software to google drive or any website. It also can keep the stored file in memory or external flash memory.

The system uses two scanners; each scanner for each side page of the book. These will not run at the same time after receiving Arduino signal, because there is a shift in the 2nd scanner‘s location. The reason of this shift is that the two scanners will cause a conflict at the head of triangular prism, and this problem would affect scanning the internal side of the page; and so the only solution would be to shift scanner 2 location and delay its running.

The programming language used is Python. This language supports scanners access, by using SANE package, and it facilitates scanners control, Python supports SANE package and supports GPIO package, so that is why it is used for programming.

As explained previously, each scanner will run separately, and both will scan separately each side page. It is preferable to write two programs for each scanner, as Python code would also execute commands; line after line. Hence the second scanner would be delayed and would run after 1.5 second of the first scanner starting.

At the start of Python code for both programs, all packages used in the code should be defined. These are:

1) SANE: used to access to the scanner.
2) GPIO: to receive Arduino signal on GPIO pin.
3) Time: this package has variable commands for timing, the most used is time. Sleep, it is similar to delay in mikro C.
4) Numpy: is the fundamental package for scientific computing with Python, and it helpful in image saving quickly comparison with other packages.

```
from __future__ import print_function
import sane
import numpy
from PIL import Image
import time
import RPi.GPIO as gpio
```

After import packages, it should define GPIO pins, the only pin that used is GPIO4, and it receive the Arduino signal that run the scanners.

```
gpio.setwarnings(False)
gpio.setmode(gpio.BCM)

gpio.setup(4,gpio.IN)
```

The next step in the python is scanner options Cano Scan Lide 120 provide grey and color scanning, and we used color for all cases, although grey scanning has less scanning time, but color scanning ensure time schedule for project needs. There is another option for scanning, which is the image's depth. It is preferred to set it for 16 bit than 8 bit, because it will affect the image resolution.

```
depth = 16
mode = 'Color'
```

After that it should select the scanner that used. It is important to know which scanner has first arrangement before setting it on the structure, the other scanner will automatically be selected as "Scanner two" in program "two".

```
devices = sane.get_devices()
print('Available devices:', devices)
dev = sane.open(devices[0][0])
dev.mode=mode
params = dev.get_parameters()
print('Device parameters:', params)
```

Now, it should divide pages between the two scanners, the best way to divide the pages, is to use two list for each scanner, scanner 1 take the odd numbers, and the scanner 2 take the even numbers, the sum of odds and evens should be the number of desired scanned pages, after determination pages numbers for scanners, it should make loop for each number in the list, for every number the scanner will start scanning if the Arduino send signal and it will use the number to name the image at saving process.

```
y=input('enter number of pages: ')
l=list(range(1,y+1))
odds = [x for x in l if x % 2 != 0]
for n in odds:
    print ('waiting')
    while (gpio.input(4) == False) :
        continue
    dev.start()
    im = dev.snap()
    im.save('test_pil%d.png'%n)
    time.sleep(6)
```

Some hints for the previous code:

1) Y is the number of total pages.
2) L is list that contain numbers from 1 to y.
3) Odds is list that contain odds number from 1 to y.
4) Gpio input (4): is the signal that came from Arduino, if the signal low, it will enter while loop, and it will not continue until the signal change, if it changed to high (True), it will go out from the loop and start scanning process.
5) Im.save('test_pil%d.png'%n): it is saving formula, %n that mean it will used loop number in name of scanned image.
6) Time. sleep(6): it should delayed 6 second until stage 1 finished, because in this stage, the scanner signal is High, if the program finish scanning and the signal still high, it will scan in stage 2, and this is wrong, so it should be to wait 6 second until changing signal, then starting new loop number and waiting the signal.

The other scanner code is not differs from the first one, except that choosing scanner will be automatically as explained previously, and the number that used in the loop is even numbers, in addition to there is delay for 1.4 second before scanner two running because scanner 2 position shifted on the structure.

```
y=input('enter number of pages: ')
l=list(range(1,y+1))
evens = [x for x in l if x % 2 == 0]
for n in evens:
    print ('waiting')
    while (gpio.input(4) == True) :
        continue
    time.sleep(1.4)
    dev.start()
    im = dev.snap()
    im.save('test_pil%d.png'%n)
    time.sleep(6)
```

The first scanner will scan the left side of the book and it will save name images for odd number, and the second scanner will scan the right side and it will save name of images for even numbers, after scanning all pages, the next stage is to convert all images to PDF files, and merging it, as shown in the following code.

```
import PIL.Image
from pyPdf import PdfFileWriter, PdfFileReader

y=input('enter number of pages: ')
l=list(range(1 , y+1))
for n in l:
        filepath = 'test_pil%d.png'%n
        newfilename = 'out%d.pdf'%n
        im = PIL.Image.open(filepath)
        im.save(newfilename, "PDF", quality=100)


for n in l:
                input= PdfFileReader(file("out%d.pdf"%n, "rb"))
                output.addPage(input.getPage(0))


outputStream = file("final.pdf", "wb")
output.write(outputStream)
outputStream.close()
```

After finishing first scanner program and second scanner program, the system execute converting and merging program, in the code introduction, it import the packages that used in the code, then it should enter the number of pages, to use it in two loop, first loop to convert each image to PDF file, second loop is used to merging PDF files together, finally, it will produce the final PDF file, and it will save in default path in Raspberry Pi.

7

Chapter Seven

Validation Result, Recommendations and Manual of the System

7.1 Validation Result of the system.

7.2 Recommendation for the System.

7.3 Manual for the System Operation

7.1 Validation Result of the system.

In order to systematically demonstrate that the specifications and needs for system have been met, by conducting the following empirical tests after finalizing the design of the system.

Scanning Rate and Failure Rate

In order to validate our pages per hour and failures per hour requirements, first by determination the speeds for scanning cycle. By determination the necessary saddle speed for scanning operation to be about 2.2cm /second, this is dictated by the scans pixel density. After completing scanning, by running the saddle at 13.5 cm/second to finish the cycle. This left us with a cycle time less than 24 seconds and this meet the needs and requirement.

Then by running the scanner cycle for four different books, these are shown in Table 7.1. Every book was tested for 30 cycle.

Table 7.1 Tests for four book

Book	Description	Success of page flipping for 30 cycle
1	Thick paper, good quality, hard cover, standard size	100 %
2	Thin paper, good quality, soft cover, standard size	100 %
3	Thick paper, good quality, soft cover, small size	100 %
4	Thin paper, good quality, soft cover, small size	97 %

Book Specifications

By making many tests on different book sizes and we validate that the system meet the requirements, it was successful to scan books with 3 KG weight and less, also by the tests it can be scan the A4 page and less than.

Price:

The goal for the scanner price was to be under $1,500. After adding up the cost for the scanner components as well as the extra parts to operate the scanner. The final cost is validated to be under $1500.

Friendly User Interface

This point was achieved by putting (on-off, Reset and emergency) switches, there is a monitor that shows the scanned pages and the final PDF files and there is a mouse and keyboard was connecting to the Raspberry PI.

7.2 Recommendation for the System.

After completing our system design, there are many points of recommendations to improve the design going forward in the future.

1) It recommended to use Acrylic with 5 mm thickness instead of 4 mm, to make the system more strong.
2) It recommended to use faster scanners such as CanoScan LIDE 220, instead of CanoScan LIDE 120, that will increase the speed of the system and will increase the number of scanned pages in hour.
3) It recommended to use different vacuum device to reduce the noise level.
4) It recommended to use double feed detection ultrasonic sensor, that will allows the system to be more accurate in page detection.
5) It recommended to use OCR to convert the images into texts.

7.3 Manual for the System operation

In previous chapter, it explained the programming, wiring and testing each system alone, the final assembly of the system should has steps to running on, because each subsystem has running conditions, at the beginning, it should turn on Raspberry Pi, because it take 5 minute to be ready, turning it on does not include scanners code, the code should be run manually, until Raspberry Pi be ready, it can be running on the power supply for motors, Arduino and vacuum cleaner.

If all subsystems were ready and turned on, it can be to start running the system, first step , it should to run the Python code for scanners by typing python mma1.py and python mma2.py in terminal window , the code need to input the required pages to scanned , then it will wait the Arduino signal for running the scanners , on the other hand , Arduino wait start signal from start and stop switch , if it change to start , the saddle will moved to starting point and stop to run the scanners for first time, first run for scanners will not take in consideration, because it takes more time at first run from next scanning operations and its scanned images will not include in the final PDF book file , as explained in previous sections.

After finishing first running of scanners , it can be start the 4 stages which explained in Arduino code , each stage has operation conditions , it can be stop any stage by stop switch , and it also can be active by start switch , after finishing the 4 stages , the system will be finished 1 cycle and it will be scanned 2 pages.

After scanning all required pages , the python code will automatically close , saddle moving should be stopped by stop switch , next step is to merge all images to PDF file , merging operation has a single code , because running 3 programs may affect overload on Raspberry Pi CPU, and it also may cause the scanning operations slower and it should run alone without running another programs, so it is better if run the merging program after stopping scanning operations, it can be run it by typing python cam.py in terminal , the program will take few seconds to convert images to PDF files and then merging all PDF fill into the final book PDF file.

References:

[1]Teach ICT Company, Input Device Scanner,2002.website

http://www.teach ict.com/gcse_new/computer%20systems/input_devices/miniweb/pg8.htm [2]

Daniel Reetz, Book Scanner From A Cardboard Box, 2009.

Website http://www.instructables.com/id/Bargain-Price-Book-Scanner-From-A-Cardboard-Box

[3] Daniel Reetz, Hardware Store Scanner, 2010.

Websitehttps://www.diybookscanner.org/forum/viewtopic.php?f=1&t=333

[4] Daniel Reetz, A DIY Book Scanner, Archivist Quill Book Scanner, 2011.

Websitehttp://www.makezine.com/projects/make-41-tinkering-toys/diy-book-scanner

[5] Treventus Mechatronics , ScanRobot 2.0 MDS, 2014.

Website http://www.sycteam.com.mx/scanner/hojasdatos/ScanRobot.pdf

[6] 4DigitalBooks ,4DigitalBooks - the only real full automatic book scanners,

2014. Website http://www.4digitalbooks.com/

[7]Qumsiyeh, D, Prism Scanner The open-source automatic book scanner,2012.

Website http://prismscanner.org/

[8]M.Margolis,*Physical output, in"*Arduino Cookbook",S. Wallace and B. Jepson,eds., O'Reilly,United States, 2012, pp. 289-326.

[9]Canon,Canoscan LIDE 220,2014. Website:

https://www.usa.canon.com/internet/portal/us/home/support/details/scanners/photo-scanner/canoscan-lide-220

[10]M.Banzi," Getting Started with Arduino", O'Reilly, Cambridge, 2009.

[11]E. Upton, and G. Halfacree, "Raspberry Pi User Guide", Wiley, United Kingdom,2012.

[12]J.Sobota, R. Piˇsl, P. Balda, and M. Schlegel , Raspberry Pi and Arduino boards in Control Education,*IFAC Proceedings, vol.*46, no.17 ,2013, pp.7-12.

[13] J. Fraden, "Handbook of Modern Sensors", Springer-Verlag, Berlin,2010.

[14] W. McKinney, Python for Data Analysis, O'Reilly, United States, 2012.

Appendix A

Datasheets of Components

A.1 Canoscan LIDE 120 Datasheet

A.2 Nema 23 Datasheet

A.3 TCRT5000 Sensor Datasheet

A.1 CanosSan LiDE 120 Datasheet

Canon

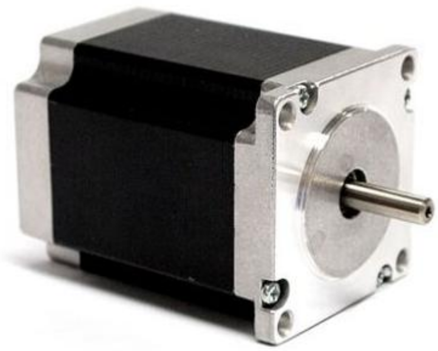
Delighting You Always

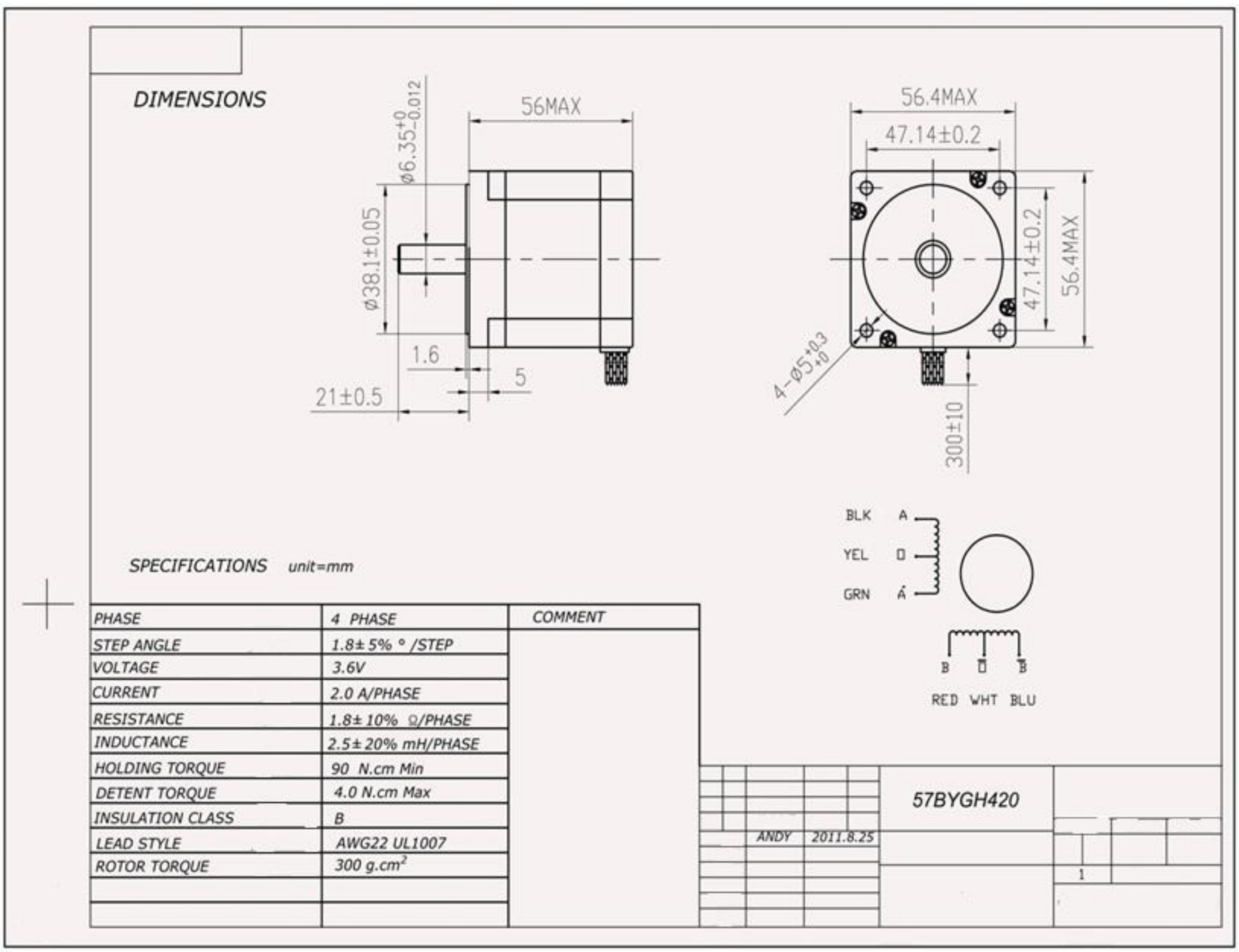
Specification

Model		**CanoScan LiDE 120**
Scanner type		Flatbed
Optical resolution		2,400 x 4,800 dpi
Scanner element		CIS
Light source		3 Color LED
Scanning bit depth	Color	48 bit input, 48 bit or 24 bit output
	Grayscale	16 bit input, 8 bit output
Preview speed[1]		Approx. 14 sec
Scanning speed[2]	Color	Approx. 16 sec (A4 / 300 dpi)
Interface		USB 2.0 High speed
Maximum document size		A4/ LTR, 216 x 297 mm
Scanning buttons (EZ buttons)		4 buttons (PDF, Auto Scan, Copy, Send)
A3 / B4 output size		Support
Gutter shadow correction		Support
Scan to PDF		Multi pages & password protection function
Power Supply		Power & connection via USB cable
Dimension (W x D x H)		250 x 370 x 40mm
Weight		Approx. 1.6 kg
Operation system		Windows 8.1, 8, 7 , Vista SP1 / SP2 , XP SP3
		Mac OS X 10.6.8 – 10.9.x[3]
Warranty		1-year carry in warranty

All data is based on Canon's standard testing methods. Specifications are subject to change without notice.

A.2 Nema 23 datasheet





SPECIFICATIONS unit=mm

PHASE	4 PHASE	COMMENT
STEP ANGLE	1.8± 5% ° /STEP	
VOLTAGE	3.6V	
CURRENT	2.0 A/PHASE	
RESISTANCE	1.8± 10% Ω/PHASE	
INDUCTANCE	2.5± 20% mH/PHASE	
HOLDING TORQUE	90 N.cm Min	
DETENT TORQUE	4.0 N.cm Max	
INSULATION CLASS	B	
LEAD STYLE	AWG22 UL1007	
ROTOR TORQUE	300 g.cm^2	

A.3 TCRT5000 Sensor datasheet

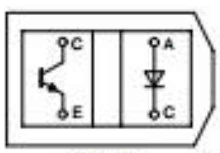
VISHAY.

TCRT5000, TCRT5000L

Vishay Semiconductors

Reflective Optical Sensor with Transistor Output

19156_2



Top view 19156_1

FEATURES

- Package type: leaded
- Detector type: phototransistor
- Dimensions (L x W x H in mm): 10.2 x 5.8 x 7
- Peak operating distance: 2.5 mm
- Operating range within > 20 % relative collector current: 0.2 mm to 15 mm
- Typical output current under test: $I_C = 1$ mA
- Daylight blocking filter
- Emitter wavelength: 950 nm
- Lead (Pb)-free soldering released
- Compliant to RoHS directive 2002/95/EC and in accordance to WEEE 2002/96/EC

e2

RoHS COMPLIANT

DESCRIPTION

The TCRT5000 and TCRT5000L are reflective sensors which include an infrared emitter and phototransistor in a leaded package which blocks visible light. The package includes two mounting clips. TCRT5000L is the long lead version.

APPLICATIONS

- Position sensor for shaft encoder
- Detection of reflective material such as paper, IBM cards, magnetic tapes etc.
- Limit switch for mechanical motions in VCR
- General purpose - wherever the space is limited

PRODUCT SUMMARY

PART NUMBER	DISTANCE FOR MAXIMUM CTR_{rel} (1) (mm)	DISTANCE RANGE FOR RELATIVE I_{out} > 20 % (mm)	TYPICAL OUTPUT CURRENT UNDER TEST (2) (mA)	DAYLIGHT BLOCKING FILTER INTEGRATED
TCRT5000	2.5	0.2 to 15	1	Yes
TCRT5000L	2.5	0.2 to 15	1	Yes

Notes

(1) CTR: current transfere ratio, I_{out}/I_{in}

(2) Conditions like in table basic charactristics/sensors

ORDERING INFORMATION

ORDERING CODE	PACKAGING	VOLUME (1)	REMARKS
TCRT5000	Tube	MOQ: 4500 pcs, 50 pcs/tube	3.5 mm lead length
TCRT5000L	Tube	MOQ: 2400 pcs, 48 pcs/tube	15 mm lead length

Note

(1) MOQ: minimum order quantity

ABSOLUTE MAXIMUM RATINGS (1)

PARAMETER	TEST CONDITION	SYMBOL	VALUE	UNIT
INPUT (EMITTER)				
Reverse voltage		V_R	5	V
Forward current		I_F	60	mA
Forward surge current	$t_p \leq 10$ µs	I_{FSM}	3	A
Power dissipation	$T_{amb} \leq 25$ °C	P_V	100	mW
Junction temperature		T_j	100	°C

Appendix B

Materials Properties

B.1 Acrylic Properties

B.1 Acrylic Properties

Physical Properties of Acrylite® FF Acrylic Sheet

Property[a]		ASTM Method	Typical Value (0.250" Thickness)[b]
Mechanical	Specific Gravity	D 792	1.19
	Tensile Strength	D 638	10,000 psi (69 M Pa)
	Elongation, Rupture		4.5%
	Modulus of Elasticity		400,000 psi (2800 M Pa)
	Flexural Strength	D 790	17,000 psi (117 M Pa)
	Modulus of Elasticity		480,000 psi (3300 M Pa)
	Compressive Strength (Yield)	D 695	17,000 psi (117 M Pa)
	Impact Strength Izod Milled Notch	D 256	0.4 ft. lbs/in. of notch (21.6 J/m of notch)
	Rockwell Hardness	D 785	M-93
	Barcol Hardness	D 2583	48
Optical	Refractive Index	D 542	1.49
	Light Transmission, Total	D 1003	92%
Thermal	Forming Temperature	—	Approx. 300°F (149°C)
	Deflection Temperature under load, 264 psi	D 648	195°F (91°C)
	Vicat Softening Point	D 1525	220°F (105°C)
	Maximum Recommended Continuous Service Temperature	—	160°F(c) (71°C)
	Coefficient of Linear Thermal Expansion	D 696	0.000040 in/in - °F (0.000072 m/m - °C)
	Coefficient of Thermal Conductivity	Cenco-Fitch	1.3 BTU/(Hr) (Sq. Ft.) (°F / in.) (0.19 w/m•K)
	Flammability, Burning Rate (0.125" thickness)	D 635	1.0 in/min. (25 mm/min.)
	Self Ignition Temperature	D 1929	850°F(455°C)
	Specific Heat @ 77°F		0.35 BTU/(lb.) (°F) (1470J/Kg•K)
	Smoke Density Rating	D 2843	4.8%
Electrical	Dielectric Strength Short Time (0.125")	D 149	430 volts/mil (17 KV/mm)
	Dielectric Constant 60 Hertz 1000 Hertz 1000000 Hertz	D 150	 3.6 3.3 2.8
	Dissipation Factor 60 Hertz 1000 Hertz 1000000 Hertz	D 150	 0.06 0.04 0.02
	Volume Resistivity	D 257	10^{16}ohm-cm
	Surface Resistivity	D 257	10^{15}ohms
Water Absorption 24 hrs @ 73°F		D 570	0.2%
Odor		—	None
Taste		—	None

NOTES: (a) Typical values; should not be used for specification purposes.
(b) Values shown are for 0.250" thickness. Some values will change with thickness or pigmentation.
(c) It is recommended that temperatures not exceed 160°F for continuous service, or 190°F for short intermittent use.