

**DESIGN AND DEVELOPMENT OF A WHEEL BALANCING MACHINE FOR
THE LOCAL MARKET**

A Final Year Project Report

Presented to

SCHOOL OF MECHANICAL & MANUFACTURING ENGINEERING

Department of Mechanical Engineering

NUST

ISLAMABAD, PAKISTAN

In Partial Fulfillment
of the Requirements for the Degree of
Bachelor of Mechanical Engineering

by

Waqar Rafique

Muhammad Asjad

Saad Bin Asad

Muhammad Hannan Ayub

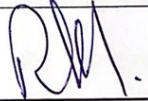
May 2024


EXAMINATION COMMITTEE

We hereby recommend that the final year project report be prepared under our supervision by:

WAQAR RAFIQUE	331699.
MUHAMMAD ASJAD	356939.
SAAD BIN ASAD	333688
MUHAMMAD HANNAN AYUB KHAN	335935

Titled: "DESIGN AND DEVELOPMENT OF WHEEL BALANCING MACHINE FOR LOCAL MARKET" be accepted in partial fulfillment of the requirements for the award of BACHELOR OF MECHANICAL ENGINEERING degree with grade ____

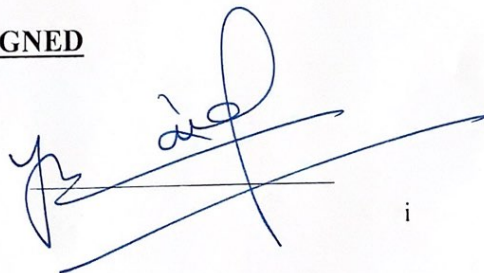
Supervisor: Riaz A. Mufti, Dr. (Professor)	 Dated: 11/6/24
Committee Member:	 Dated:
Committee Member:	 Dated:


(Head of Department)

20/6/2024
(Date)

COUNTERSIGNED

Dated: 20/6/24


i

ABSTRACT

To improve accuracy and user convenience, this report investigates the creation and application of a dynamic wheel balancing device coupled with a mobile application for the automotive sector. When it comes to addressing dynamic imbalances brought on by things like tire wear, altered road conditions, and variations in vehicle speed, traditional static wheel balancing techniques frequently fall short. Thus, a more sophisticated and flexible method of wheel balancing is required. The suggested dynamic wheel balancing device precisely detects dynamic imbalances and offers the best corrective actions by utilizing sensor technologies and real-time data processing. Furthermore, by integrating a mobile application, users can remotely monitor the balancing process, access diagnostic reports, and receive maintenance alerts, all of which improve user experience and streamline operational efficiency.

ACKNOWLEDGMENTS

As already mentioned, the manufacturing information was very difficult to extract from the Internet. As a result, we had to rely on the currently available machines we were studying and the people around us for technical knowledge. First, we would like to thank our supervisor Dr. Riaz Ahmed Mufti for guiding us through the course of this project. We would also like to thank our senior and mentor Mr. Sohaib, whose experience and guidance were of immense help during the process of sensor selection, controller selection, and development of the complete electrical system of the machine.

ORIGINALITY REPORT

ORIGINALITY REPORT

5%	4%	1%	4%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	Submitted to Higher Education Commission Pakistan Student Paper	3%
2	Submitted to University of Glamorgan Student Paper	1%
3	slideplayer.com Internet Source	<1%
4	Submitted to CSU Northridge Student Paper	<1%
5	Submitted to University of Huddersfield Student Paper	<1%
6	dspace.library.uvic.ca Internet Source	<1%
7	scholars.indstate.edu Internet Source	<1%
8	www.tib.org.uk Internet Source	<1%
9	5dok.org Internet Source	<1%

10	researchonline.federation.edu.au Internet Source	<1%

TABLE OF CONTENTS

ABSTRACT.....	ii
ACKNOWLEDGMENTS.....	iii
ORIGINALITY REPORT.....	iv
LIST OF TABLES.....	vi
LIST OF FIGURES.....	vii
ABBREVIATIONS.....	Error! Bookmark not defined.
NOMENCLATURE	Error! Bookmark not defined.
CHAPTER 1: INTRODUCTION	10
CHAPTER 2: LITERATURE REVIEW	13
CHAPTER 3: METHODOLOGY	18
CHAPTER 4: RESULTS AND DISCUSSIONS.....	38
CHAPTER 5: CONCLUSION AND RECOMMENDATION.....	72
REFERENCES:	74
APPENDIX I: TITLE OF APPENDIX I.....	76

LIST OF TABLES

Table 1: Literature Review.....	17
---------------------------------	----

LIST OF FIGURES

Figure 1 Rotating Assembly.....	12
Figure 2 Isometric View.....	19
Figure 3 & 4 Views	20
Figure 5 & 6 Frame.....	22
Figure 7 Base Plate.....	23
Figure 8 & 9	23
Figure 10 Shaft Assembly.....	24
Figure 11	24
Figure 12, 13, 14	25
Figure 15.....	26
Figure 16, 17.....	27
Figure 18, 19.....	28
Figure 20.....	29
Figure 21 Belt Selection.....	33
Figure 22 Circuit 1.....	36
Figure 23 Circuit 2.....	36

Figure 24 STM32F407.....	37
Figure 25, 26.....	41
Figure 27 Mobile Application.....	48
Figure 28 Relay Module.....	49
Figure 29 & 30	51
Figure 31.....	52
Figure 32-36 Stress Analysis.....	63
Figure 37-41 Modal Analysis.....	67

NOMENCLATURE

r	radius of wheel
W	width of wheel
L	distance of wheel from the reference plane
e	radius of unbalance mass on the rotor
G	Center of gravity
F	Force
i	terms for total unbalance in the rotor
b_1	balance mass introduced in plane 1
b_2	balance mass introduced in plane 2
ω	rotational speed in r.p.m
m_c	correction mass
m	Mass
M_y	moment in y axis
M_x	moment in x axis
F_x	force in x axis
F_y	force in y axis
V_c	vertical imbalance couple
H_f	horizontal imbalance force
V_f	vertical imbalance force
H_c	horizontal imbalance couple

CHAPTER 1: INTRODUCTION

Problem Statement:

In the local market, it is critical to balance rotating components with accuracy and efficiency. Nonetheless, there are still a few affordable, regionally-specific balancing options available. For small and medium-sized businesses in the area, there is a notable deficiency in the availability of balancing machines that are both economically viable and of the highest technological caliber.

This project focuses on the design and development of a wheel balancing machine, tailored to the requirements of the local Industry. The problem at hand was to develop a machine that had a completely modular electronic system and design.

Working Principles

There are two different kinds of wheel balancing machines that are employed: the dynamic type and the static type. The static balancing machine focuses primarily on the unbalanced forces acting on the wheel, whereas the other machine operates on different principles. In the case of static balancing, single-plane balancing takes place. Static balance also applies to things in motion as well for this case, and the requirement for the static balance is that the sum of all the forces on the moving system is zero. For static balancing, the assembly being balanced is relatively short in the axial direction as compared to the radial direction.

It is also referred to as a two-plane balance when discussing dynamic balancing. Any assembly that is longer in the axial direction than the radial direction needs to have dynamic

balance. That an object is a possibility. It is necessary to either add or remove the appropriate amount of weight at the proper angular locations in two correction planes spaced apart by a certain amount of shaft length to correct the dynamic imbalance. The inner and outer edges of the wheel rim are the proper places for an automobile wheel.

Because an automotive wheel assembly is made of homogeneous metal and has a uniform geometry, it is expected to be close to being in a balanced state when it is manufactured. In contrast, a tire is made of a composite material that varies in density and distribution and is composed of synthetic rubber elastomer and fabric cord or metal wires. This is why we perform wheel balancing on automotive wheel assemblies. When dynamic balancing is used, the assembly that must be balanced is placed on an axle or mandrel that is supported by bearings inside the balancer. The suspension system that holds the bearings is mounted on a transducer, which is an electronic device that changes the form of energy and converts physical force into an electric signal.

A shaft encoder mounted on a common force transducer provides the reference signal, which is used to calculate the peak magnitude and phase angle of the peak about a reference signal. Common force transducers contain piezoelectric crystals that will deliver a voltage proportional to the force applied and this voltage would be further amplified. Once every revolution, in the same angular location, the shaft encoder pulses a brief electrical pulse that prompts the computer to start processing the force signal. Additionally, the encoder will produce a significant number of additional pulses (roughly 1024) evenly spaced around the shaft circumference. These pulses are used to initiate the recording of

each data sample from the transducers at the same location around the shaft, and with the aid of an electronic counter, they will enable us to determine the shaft velocity.

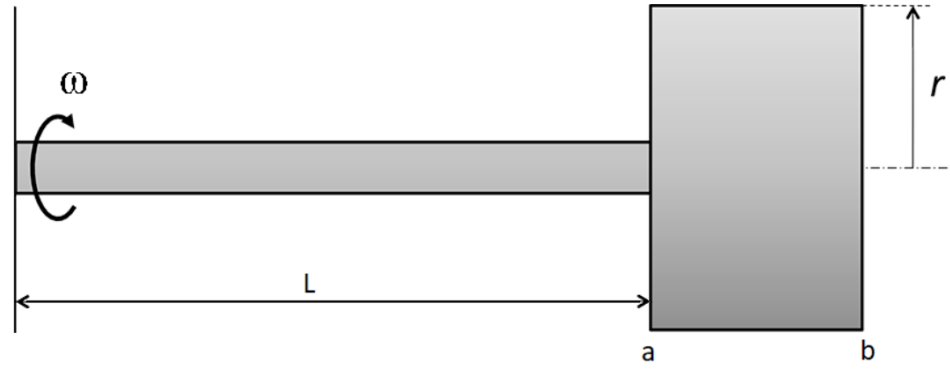


Figure 1: Rotating assembly

The mathematical equations used in the machine are as follows:

$$\begin{aligned} \sum M_y = 0 & \quad m_a r \omega^2 l_a \sin \theta_a + m_b r \omega^2 l_b \sin \theta_b + V_c = 0 \\ \sum M_x = 0 & \quad m_a r \omega^2 l_a \cos \theta_a + m_b r \omega^2 l_b \cos \theta_b + H_c = 0 \\ \sum F_x = 0 & \quad m_a r \omega^2 \cos \theta_a + m_b r \omega^2 \cos \theta_b + H_F = 0 \\ \sum F_y = 0 & \quad m_a r \omega^2 \sin \theta_a + m_b r \omega^2 \sin \theta_b + V_F = 0 \end{aligned}$$

CHAPTER 2: LITERATURE REVIEW

In this section, the detail of the previous study is presented for the dynamic wheel balancing machine is provided which ranges from different practices used in motor selection, belt selection, and different types of reinforcements for the frame ensuring the structural integrity of the machine. In the table below, the different practices are enlisted which are explained in detail here.

The motor used in the first machine was a single-phase A.C motor of 0.5 hp. and 1320 rpm. It was needed to use a motor with less horsepower rating because of the National Electrical Manufacturers Association (NEMA) minimum sheave (Pulley) requirements. NEMA is an association that has issued a range of minimum recommended sheave diameters for electric motors [16]. For a 1 Hp motor, the minimum recommended sheave diameter is 2.4 inches, so if the motor of 1 Hp and 1400 rpm we would require a large pulley of 16.8 inches, which would be not feasible. Another solution to this problem is that we use a large motor of 2 HP but with a motor controller (Variable frequency drive). By using a VFD, we can rotate the motor at a lower rpm than the rated rpm, so this reduces our speed ratio and thus reduces the need for larger diameter pulleys. By keeping these factors in mind, the motor that was selected was an A.C motor of 2 hp and 1425 rpm for our drive mechanism along with a VFD compatible with this motor.

For our sensors, in the case of a dynamic wheel balancing machine, two areas are of key interest. The first area of interest in this case is the area of the machine where the wheel or the assembly that is about to be balanced is added.

In this area, two sensors will be located, and it will be located such that it will be one on each side. It was found that in this area of the machine, optical sensors were used quite frequently. The common types of optical sensors that can be used are photoelectric sensors and laser sensors. Photoelectric sensor will use a light source and detector and the sensor emits a light that will be interrupted with the help of the spokes of the rim or the wheel. The second option will be the use of laser sensors, whereby a laser beam detects the position of the wheel. Laser light is reflected off the wheel and is detected by a receiver. Laser sensors were found to be used in high-end balancing machines. Sensors are used in the rotating shaft, and these are typically force sensors which include our piezoelectric sensors, strain gauge, and capacitive sensors. Strain gauges are thin wires or films used for measuring the deformation while the piezoelectric sensor contains a crystal that generates an electric charge when subjected to a mechanical force. There are mainly two types of sensors that will be used on the shaft which include the radial force measuring sensors and the lateral force sensors. Both the radial force and lateral force sensors are mounted in the form of pairs to measure the difference in each force and the difference is used to calculate the force. Radial force sensors will be oriented perpendicular to the shaft. These sensors will be mounted on a metal plate attached to the machine's spindle and they can use piezo and strain gauges. Lateral force sensors will always be located at the side of the wheel mounting area and will be oriented parallel to the shaft and they may use strain gauges or inductive sensors.

In a similar approach, the sensors used are piezoelectric for measuring the imbalance force and an optical encoder for measuring the position of imbalance. The balancing machine

would contain the sensors, drive mechanism, display/App, and controller for the motor. The piezoelectric sensor is mounted under the bearing housing and a belt drive mechanism is used. A Piezo sensor is a device that gives voltage when a fluctuating force is applied to it and the piezoelectric sensor must be calibrated. To calibrate the piezo, install the piezo under the bearing housing and place a balanced rotor, and a known imbalance mass is placed at a known radius on that balanced rotor. This unbalanced mass generates an outwards centrifugal force which is transferred to the piezo sensor through the bearing housing, the maximum value of the voltage that the piezo generates due to this unbalanced mass is noted down. Now the unbalanced mass is increased and gain maximum value of voltage corresponding to this unbalanced mass is noted down. This gives us a relation between the unbalanced mass and voltage of the piezo in the form of an equation, which will be used in our program to calculate the amount of unbalanced whenever the machine is used. The drive mechanism in this case that will be used will have a motor, belt drive mechanism a shaft, and a bearing. The motor is the main component of the drive mechanism, which is driving the shaft through a belt drive mechanism. Several factors need to be considered before finalizing a motor. The motor selected should have enough starting torque to drive a stationary wheel mounted on the shaft. It should be strong enough to support varying loads on the shaft because all the wheels do not have the same inertia. What happens when we select a weak motor is that when a heavier load is applied to the shaft, the motor's speed decreases, and it is not able to rotate the wheel at the required balancing speed. This further causes the motor to draw more current in it and this high amount of current can cause problems and damage the motor's windings.

The imbalance force is detected by the two piezoelectric sensors, these piezo are placed at an angle of 90 degrees to each other and can detect both the static and dynamic imbalance and the rotary encoder will be used to measure the angular position of the shaft and rpm of the mounted wheel.

Another important aspect that can alter the course to solve this problem is the use of microcontrollers. In a range of approaches and applications, it was found that Arduino was the most common type of microcontroller implied for this purpose, but a significant issue raised for this approach is the low processing rate of the microcontroller which can cause a lag. To address this, various applications use industrial-scale microcontrollers, which range from Raspberry Pi for use with screens and STM32 for use with Bluetooth modules and without the presence of a screen which perfectly fits our use case.

Sr. No	Publisher	Title	Findings
1	Yilmaz, Emin	Microcomputer Based Instrumentation for Student Designed Wheel Balancing Machine	Described the necessary equations to calculate the required mass/force to remove the imbalance in the wheel and prototyping of a wheel balancer.
2	Gates	Industrial V-Belt Design Manual	Design and selection of V belt required for the operation of a dynamic wheel balancing machine, to transfer power from the motor to shaft assembly.
3	Atlas	Atlas WB49-2 Pro Self-Calibrating Computer Wheel Balancer	Design inspiration for the 3D Model and external frame as well as reinforcement.

Sr. No.	Publisher	Title	Findings
4	Devices, Analogue	CN0350 Data Acquisition	Basic concept for the circuit design of the DAQ for Piezoelectric sensors.
5	ST Life	STM32Fxxx Discovery Board Catalogue	Microcontroller Selection based on the requirements of the sampling rate and Bluetooth compatibility.
7	Nabar, Aneesh	A Study of Operational Variables Influencing Wheel Balancing Measurements	A study of Operational Variables affecting balancing procedure. Moreover, a study about the change in location of the piezoelectric sensor.
8	Tutorials point	React Native Tutorial	In-depth tutorials on React Native which is a JavaScript framework for building native mobile apps

Table 1: Literature Review

CHAPTER 3: METHODOLOGY

Design:

1. Introduction

Balancing rotating parts, whether a wheel or a rotor/disk, is a very intricate matter. Any sort of imbalance or fault in the assembly can be catastrophic to the balancing process and can lead to inaccuracy while balancing. For this reason, it is necessary to design the machine in a fashion that any designed component of the machine does not impact the balancing procedure negatively. This also makes it important to choose materials that are feasible economically and make no compromise on promised quality.

All these factors are kept in mind while designing and fabricating this dynamic wheel balancing machine. In the following sections, the detailed concept design as well as the fabrication and assembly process will be discussed.

2. CAD Design:

The initial design/prototype was developed using *SOLIDWORKS 2022*. Each of the major components was designed separately as a part and then joined together to form a unified assembly. The major components that were required to be designed in the CAD model were the external frame, the shaft assembly, and the support plate with the piezoelectric sensor mounting rods. The other components such as the v-belt and pulley are additional components that are not critical to the CAD

model. However, these components will be designed and selected mechanically and justified in the next section.



Figure 2: Isometric View of Assembly

- **Frame:**

The external frame was designed by using the currently available commercial wheel balancing machines as a reference. Using successive iterations, a design for the external frame was chosen. An important part of the frame is the internal reinforcement. Since the machine will be subject to mechanical vibrations, it should be reinforced to stand the course of time. Multiple simulations such as stress analyses were conducted to ensure the frame's strength and integrity. These simulations will be explained in the next sections.

- **Shaft Assembly:**

The shaft assembly is a very critical component of a wheel balancing machine. It is the component that has to be able to withstand the weight of the wheel, hence its design is very crucial. It can be divided into rotating and non-rotating parts. The design calculations will be provided in the next section. Another important part of the shaft assembly is the optical encoder teeth, which are integrated into the shaft assembly. An optical encoder is located right above the teeth of the encoder disk. The functionality of the optical encoder will be explained in the upcoming sections.

- Base/Support Plate:

The base/support plate, as the name suggests, is designed so that it can withstand the weight of the shaft assembly along with the induction motor. A rectangular-shaped steel plate with appropriate dimensions was chosen for this purpose. The thickness of the plate was chosen to be 10mm as this would provide enough support to the shaft assembly and motor.

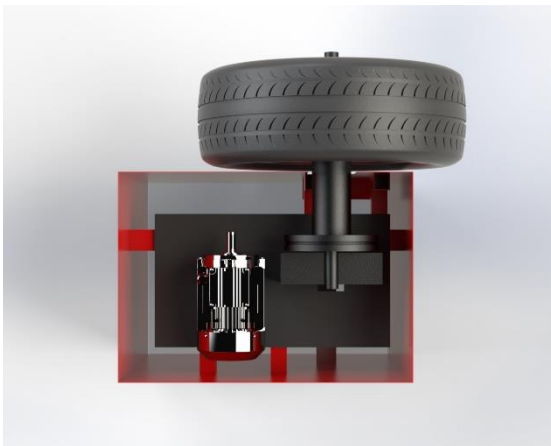


Figure 3: Top View of the Assembly

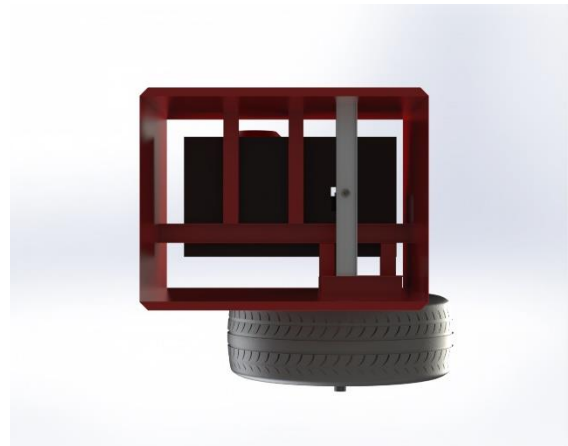


Figure 4: Bottom View of the Assembly

Model Description and Design Procedure:

1. Frame

The outer structure of the machine is a sheet metal frame. It serves the purpose of holding the components together while not being affected by any sort of forces acting on them during the machine's operation.

This design of the frame was finalized after a comprehensive Finite Element and Modal analysis, the results of which are shown in the next part of the report.

Inside the rectangular sheet metal frame are the C-shaped channels/beams which act as reinforcement members and resist the deforming forces which act on it during operation.

After necessary modifications, the available frame was ready to be used. The inspiration for modifications came from analyses conducted as well as the requirement of components to be fitted in the machine. These modifications include:

- Making a bigger slot for the rotating shaft
- Cutting the existing motor platform to replace the mounting according to our design.
- Welding an extra C-shaped channel for the placement of a piezoelectric sensor.

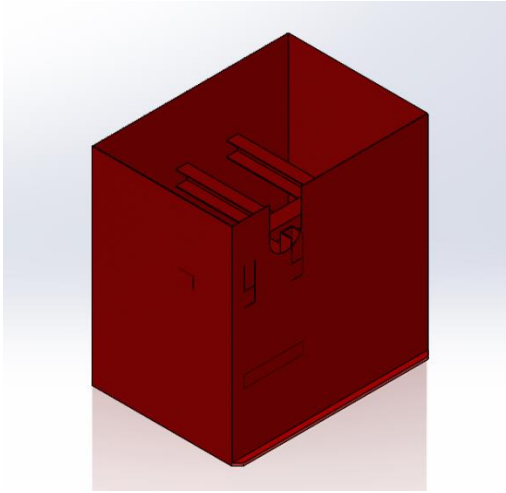


Figure 5: Isometric View of Frame

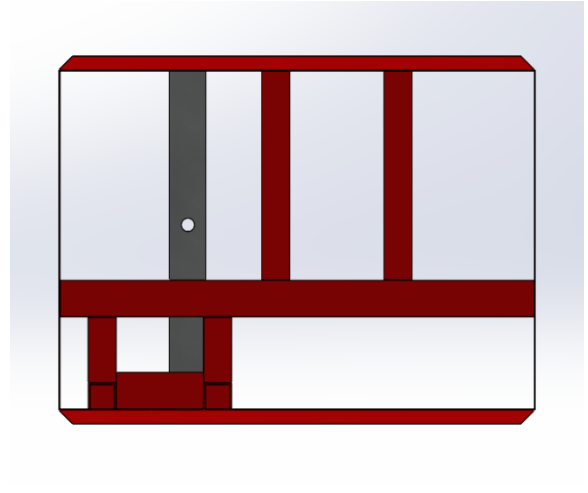


Figure 6: Top View of Frame

2. Base Plate:

The Base Plate is a 620 x 350 x 10 mm mild steel sheet. It is a major structural component of the machine, as two rotary components i.e., motor and shaft, are assembled on it.

It is mounted on the frame using M13 bolts. The required holes for bolts were drilled using a *magnetic drill press*. Moreover, 4 rectangular slots were made using *oxyfuel cutting*, to mount the motor on the plate.

Furthermore, 2x slide plates were welded onto the base plate which would hold the shaft in its place.

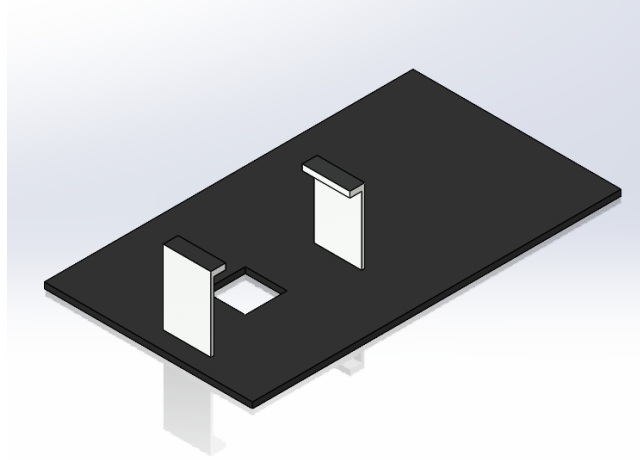


Figure 7: Base Plate

3. Shaft Assembly:

The shaft is one of the most critical components of this machine from the design perspective. The wheel to be balanced will be mounted on this shaft and rotated for 8 seconds to detect the imbalance magnitude and location on the wheel.

It is necessary that a shaft withstands not only the loads but also a variety of stresses both bending and torsional induced due to the mounted wheel on one end and pulley on the other end.

It consists of 2 main parts i.e., rotary, and stationary parts. The stationary end is attached to the piezo assembly and has the rotary shaft mounted onto it. The rotary end incorporates both the pulley at one end and the wheel and locknut assembly at the other threaded portion. Following is a detailed description of the shaft design procedure:

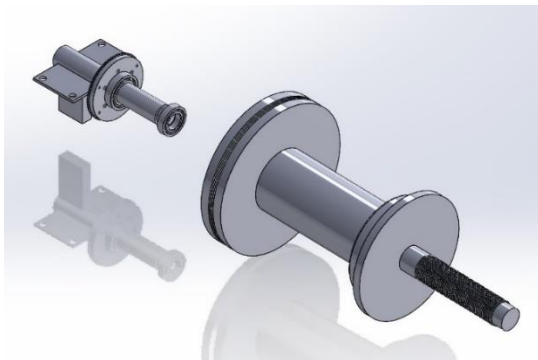


Figure 8: Exploded View

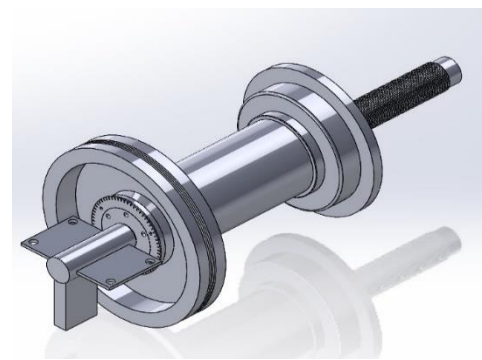


Figure 9: Isometric View



Figure 10: Shaft Assembly with sensor arrangement

Rotary Shaft Design:

Various loading conditions were considered, and a free-body diagram was drawn of the shaft which included bearing reactions and load on the shaft due to the weight of the mounted wheel and the forces induced by the belt drive mounted on the pulley.

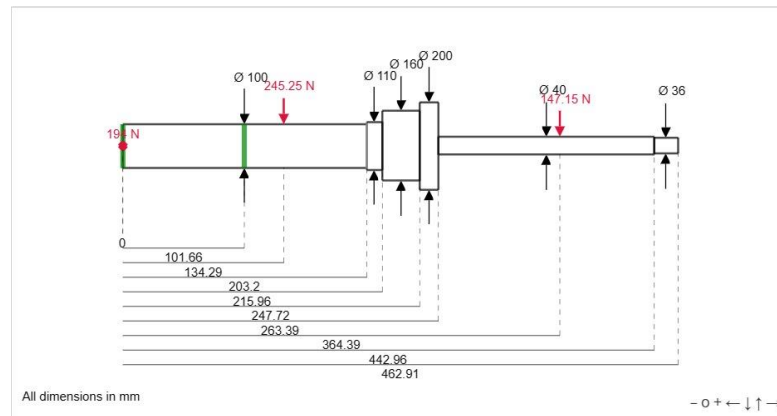


Figure 11

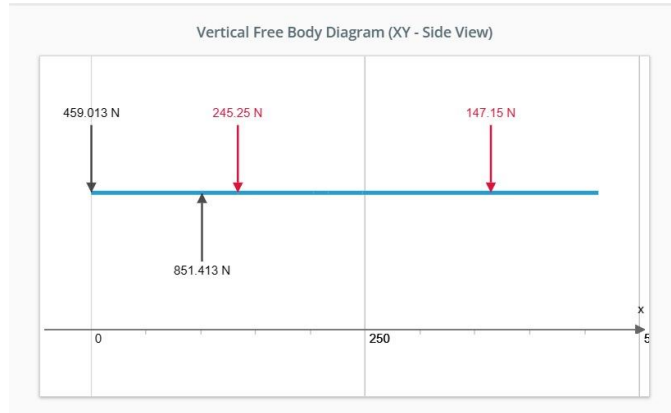


Figure 12

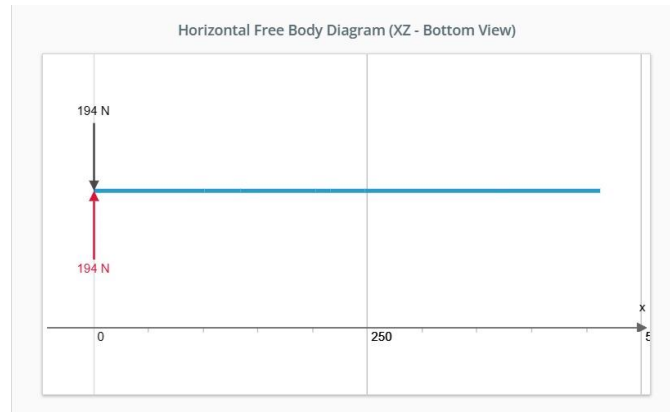


Figure 13

These reaction forces in both planes were then used to evaluate shear force and bending moment diagrams for the rotary shaft.

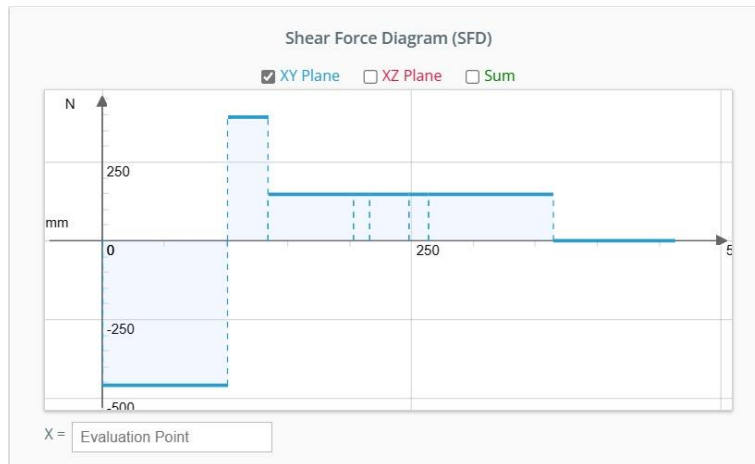


Figure 14

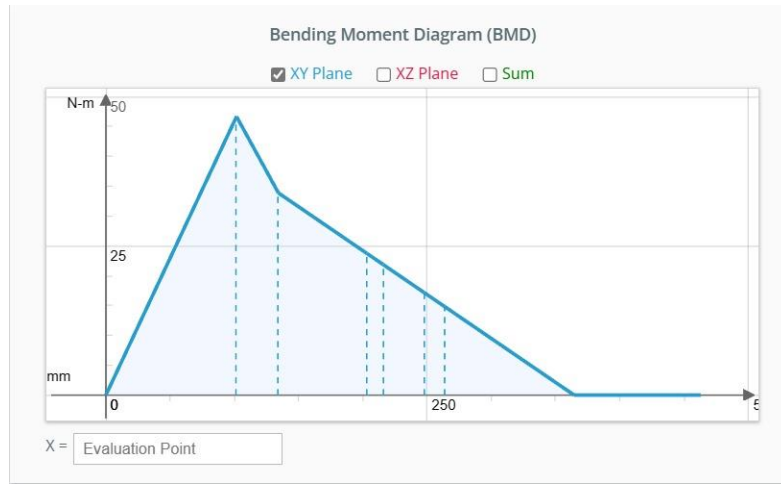


Figure 15

Using these diagrams, we can evaluate the point on the shaft that will be subjected to the maximum bending moment and torsion. We first need to find the endurance limit of the shaft.

$$S_e = k_a \times k_b \times k_c \times k_d \times k_e \times S_{ut}$$

For ultimate tensile strength, we referred to the table of 1018 Annealed Steel which was found to be 341 MPa. The coefficients for endurance limits were evaluated using the appropriate tables available in Shigley's book on Machine Design.

Using the endurance limit, we evaluate the minimum diameter of the most stressed parent by using the Goodman Criteria given below:

$$\left[16 \times \frac{10}{\pi} \left(\frac{2(K_f \times M_a)}{S_e} + \frac{(3^{\frac{1}{2}})(K_{fs} \times (T_m^2))^{\frac{1}{2}}}{S_{ut}} \right) \right]^{\frac{1}{3}}$$

Using a factor of safety of 10, which is an industrial standard, the recommended diameter was found to be 37.116 mm.

Stationary Shaft Design:

Various loading conditions were taken into account and a free-body diagram was drawn of the shaft which included bearing reactions and load on the shaft due to the weight of the mounted rotary shaft.

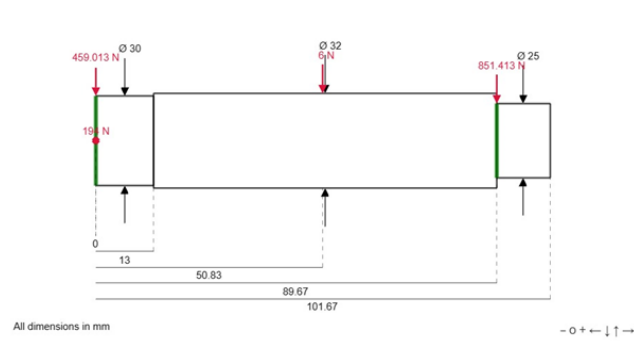


Figure 16

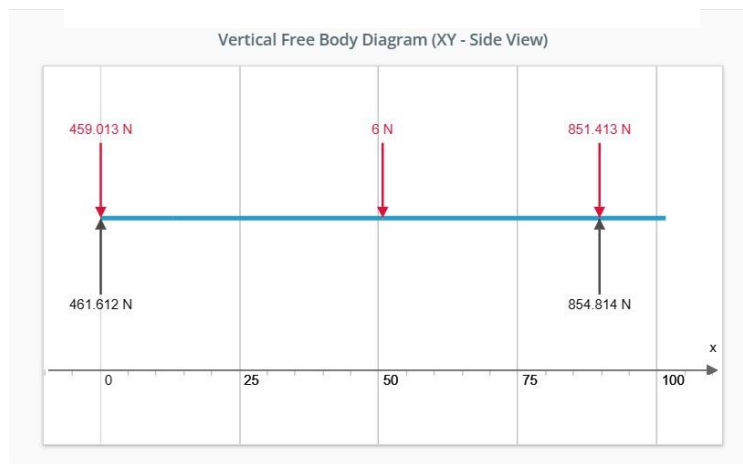


Figure 17

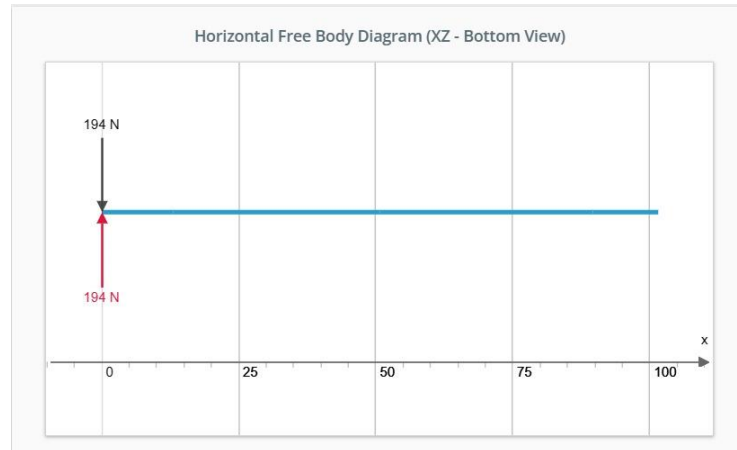


Figure 18

These reaction forces in both planes were then used to evaluate shear force and bending moment diagrams for the stationary shaft.

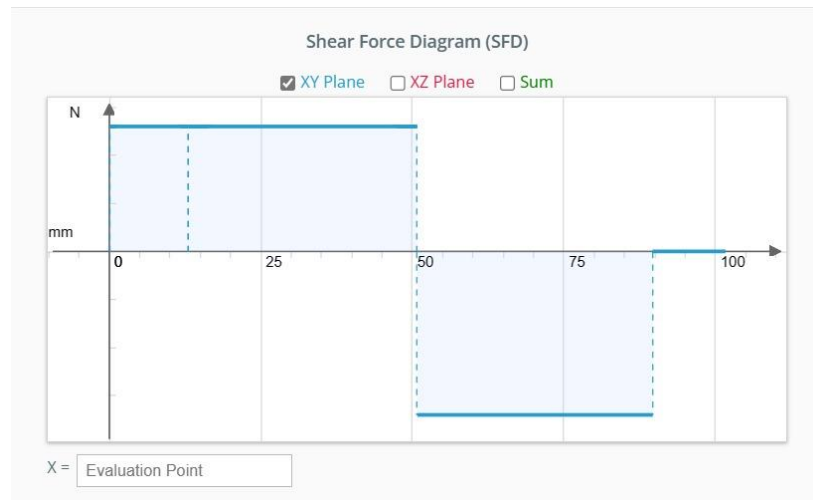


Figure 19

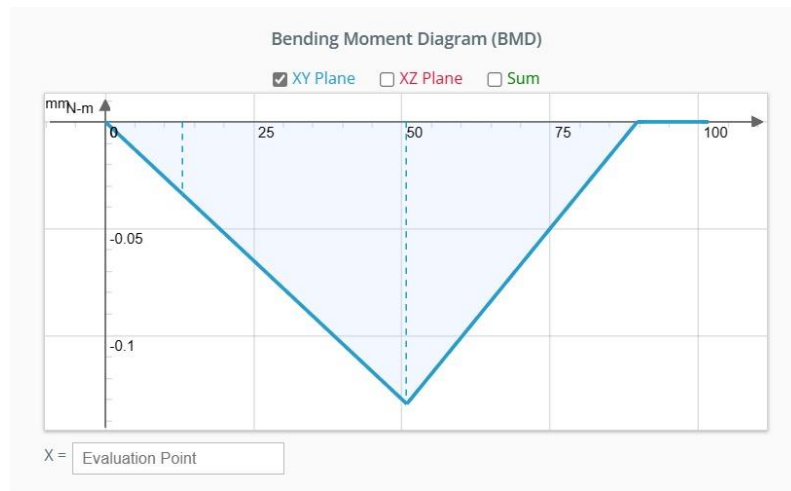


Figure 20

Using these diagrams, we can evaluate the point on the shaft that will be subjected to the maximum bending moment and torsion. We first need to find the endurance limit of the shaft.

$$S_e = k_a \times k_b \times k_c \times k_d \times k_e \times S_{ut}$$

For ultimate tensile strength we referred to the table of 1018 Annealed Steel which was found to be 341 MPa. The coefficients for endurance limits were evaluated using the appropriate tables available in Shigley's book on Machine Design.

Using the endurance limit, we evaluate the minimum diameter of the most stressed part by using the Goodman Criteria given below.

$$d = \left[16 \times \frac{10}{\pi} \left(\frac{2(K_f \times M_a)}{S_e} + \frac{(3^{\frac{1}{2}})(K_{fs} \times (T_m^2))^{\frac{1}{2}}}{S_{ut}} \right) \right]^{\frac{1}{3}}$$

Using a factor of safety of 10, which is an industrial standard, the recommended diameter was found to be 26.262 mm.

Bearing Selection

Since shafts need to be mounted onto a bearing for smooth operation and to minimize frictional losses, an appropriate bearing had to be selected for the stationary part of the shaft assembly which would also serve as support for the rotary part which is to be mounted later on. Using the free-body diagrams evaluated in shaft design, we determined the radial forces that the bearings will be subjected to. For the selection of our bearings, we referred to NSK's catalog on deep groove ball bearings. Knowing the recommended shaft diameter and the desired loading conditions we evaluated the catalog rating of the bearing using the formula below.

$$C_r = F\left(\frac{L}{10^6}\right)^{1/3}$$

Allows the use of 6005 bearing as we obtain a $C_r=522.643$ for a d_{\min} of 25 mm. For the second Bearing, the minimum diameter assumed is 30mm, $C_{req}= 891.96$ lbf is obtained. Therefore, Bearings 6005 and 6006 have been selected from the NSK catalog.

4. Drive Motor:

The drive motor is a 220-volt single-phase induction motor having a power of 0.25 kW which acts as a driver for the shaft. The shaft and motor are coupled using a V-belt, the detail of which is covered later on in this chapter.

Motor selection is an important part of the design phase. The factors that affect this phase include the required rpm, torque, power, etc. for smooth operation of the machine.

Before selecting a suitable motor for our operation, insight is needed on the output end of the belt drive mechanism. Most balancing operations are performed at 200 rpm due to the sensitivity of the piezoelectric sensor that is in contact with the shaft assembly. For this purpose, we need to ensure that the motor being selected can withstand radial loads acting on the shaft due to the pulley sheave of a belt drive mechanism and has sufficient torque to drive a wheel from its stationary position. Thus, it is viable that we refer to Cantoni Motor's catalog on General Purpose Single-Phase Induction Motors which falls under standards set on motors by ISO. So initially we used SKF Catalogue on Deep Groove Ball Bearings to estimate the minimum moment required to overcome the frictional force of the bearing and to maintain a constant speed. After evaluating the minimum required torque to drive the shaft assembly, we can evaluate the rated torque of the motor.

$$\tau = \frac{9.55 \times P}{n}$$

After evaluating the rated torque of each motor and categorizing motors which fall under this minimum torque requirement, we can also evaluate using the formula below if the applicable loads due to the pulley can be withstood by the motor shaft.

$$F_R = \frac{19600 \times P \times k}{D_k \times n}$$

Here for a v-belt.

$$k = 2.2$$

After evaluating the pulley diameter using the rated value of radial force and looking into the dimension constraints and limitations, a selection was made. We selected a 0.25 kW/0.33 HP single-phase induction motor with a rated rpm of 1100-990 rpm with IP44 protection. This is a foot-mounted motor with a frame size of 56 and a rated radial loading of 0.20 kN which was determined using the catalog.

5. Pulley and V-belt:

The system consists of 2 pulleys i.e., motor side and shaft side pulleys, and 1 V-belt which couples them.

The size of the shaft side pulley has already been calculated earlier on in this chapter. The design procedure for the rest is as follows:

It was optimal to now select the appropriate pulley size and belt length since we had input and output speeds available to us which are 1000 rpm and 200 rpm respectively. For this purpose, we consider the Gates Design Manual for Industrial V-Belts.

Design Power was evaluated initially using the formula below.

$$P_D = P \times 1.3$$

Here the constant was obtained from the service factors table available in the manual.

Using the Cross-Selection chart available in the manual we selected the J category of Micro-V belts.

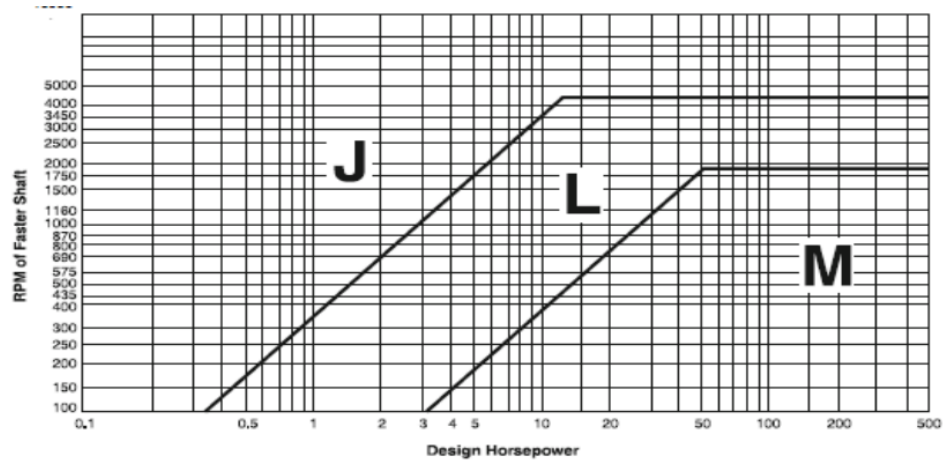


Figure 21: Belt Selection

The speed ratio was evaluated using available values.

$$S = \frac{n_i}{n_o}$$

Recommended pulley diameter was selected from the available set of diameters using the speed ratio available. Knowing the design constraints such as the center distance between the shaft assembly and the motor we can evaluate the tentative belt length.

$$TBL = 2 \times TCD + 1.57 \times (D + d) + \frac{(D - d)^2}{4 \times TCD}$$

Using tentative belt length, we can select a belt from the available list of belts in the manual. For our application and availability, the 370J belt was selected. Using the actual belt length, we reevaluated the actual center distance between the motor and the shaft assembly. $AA = F \times h$

to be 480.2 mm.

Using the manual available table of recommended pulley datum diameters for appropriate speed ratios, we made a selection of the datum diameter of driver and driven pulley which in this case turned out to be 20 mm for driven and 100 mm for driver pulley.

6. Piezoelectric Sensors Mounting Rods:

2x rods are assembled on the stationary part of the shaft on which piezoelectric sensors are later mounted to detect the imbalance force on the wheel when rotated.

These are simple aluminum rods with threads on them which enable easy assembly.

Signal Conditioning

It is necessary that vibrations are to be measured so that the issue can be successfully diagnosed. For that reason, a piezoelectric assembly was deployed in the machine to detect any fluctuation in the shaft assembly during rotation due to an imbalance in the wheel. The piezoelectric sensors employed are of the ceramic type which when subjected to force convert it to charge. Hence a circuit system had to be made to convert this charge to voltage

and then to appropriate units of force. Firstly, we need to establish a dual polarity power source to power the signal conditioning circuit which doesn't have any noise due to rectification from AC to DC.

The circuit consists of a center-tapped transformer which will step down the voltage to 9V AC from 230V AC at 50Hz. The output from two 9V AC mains will be passed through 4 1N4007 diodes for rectification which are in parallel with 4 noise eliminator ceramic capacitors of capacitance 0.1 μF each. This forms the complete rectifier unit with a noise eliminator.

Now a filter unit needs to be formed which consists of two electrolytic capacitors each of capacitance 1000 μF each in parallel with both outputs. Here larger the capacitance larger the filtering capacity of the circuit.

A regulator circuit is also required as voltages may fluctuate from the AC source which in return will fluctuate the DC voltage output. This consists of LM7812 which is a positive polarity voltage regulator and LM7912 which is a negative polarity voltage regulator. Both voltage regulators perform similar functions for different polarities of voltage.

For reverse voltage protection, two diodes 1N4007 have been installed which also consist of two electrolytic capacitors of capacitance 100 μF and two ceramic capacitors of capacitance 0.1 μF both across each output to ensure stable operation.

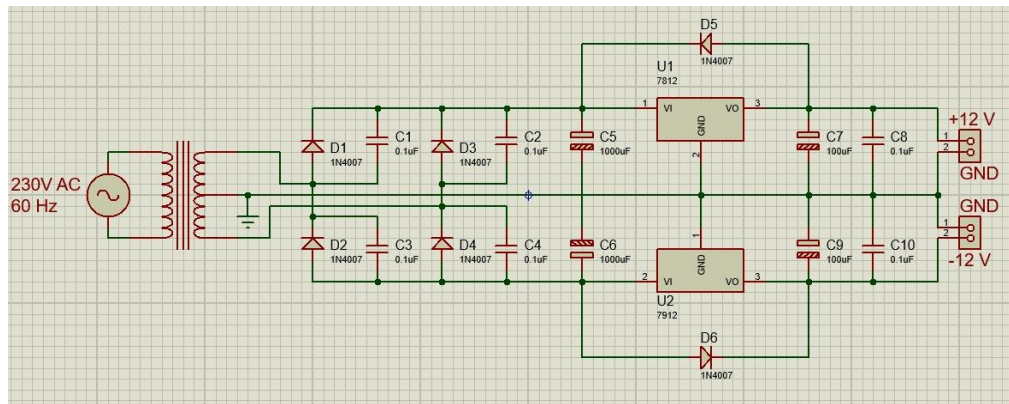


Figure 22: Circuit 1

The output of this circuit is then used to power the signal conditioning circuit which consists of 2 1 nF ceramic disc capacitors in parallel with a resistor of resistance of 10 Mohm which serves to convert the charge of the piezoelectric sensor to voltage which then passed through the CA3140EZ IC which is an operational amplifier and serves to amplify the converted voltage so that It can be detected by the microcontroller which in our case is the STM32F407.

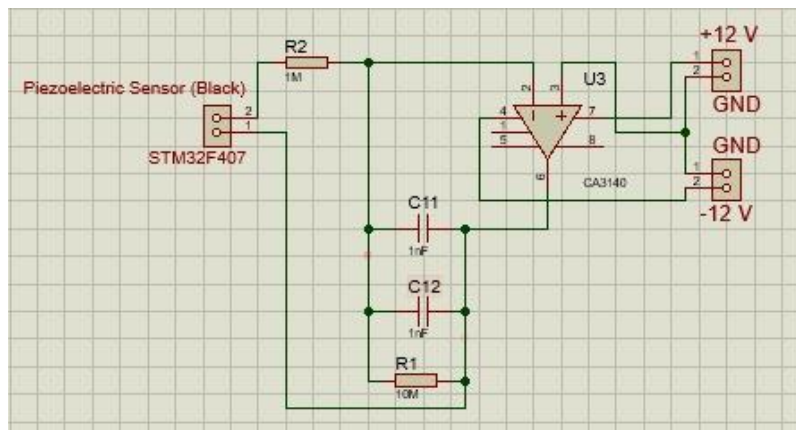


Figure 23: Circuit 2

Selection of Microcontroller

For the signal conditioning circuit, an appropriate microcontroller has to be selected to process the output signal of the piezoelectric sensor. Since 2 Piezoelectric sensors are present which also comprise an additional photo interrupter and an external clock to negate the phase change in all 3 signals, a microcontroller that has a sufficient number of ADC channels needs to be selected which in this case should be 4 or more. Similarly, since readings need me taken across the entire parameter of the wheel at 200rpm, a microcontroller needs to be selected that has a sampling rate of greater than 12000 samples per second. These needs were all catered to in our microcontroller selection which in our case is STM32F407. According to the provided brochure, this microcontroller has 8 ADC channels and has a sampling frequency of 11000000 samples per second. These specifications are more than enough to cater to our needs and thus make this microcontroller an appropriate choice for our data acquisition system.

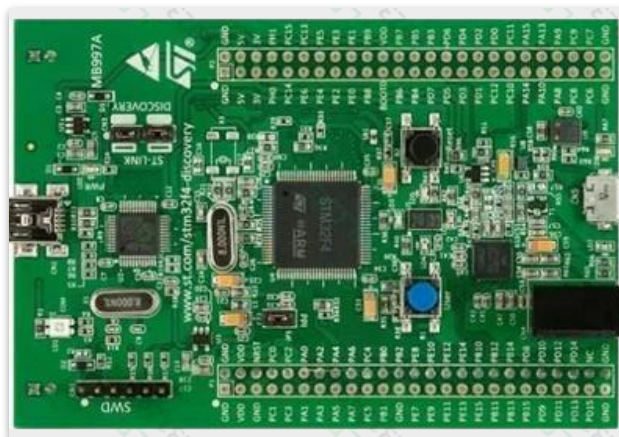


Figure 24: STM32F407

Fabrication and Assembly:

Frame:

The outer frame, as discussed in the previous chapter was modified as per the requirements of the developed design.

For this purpose, the older motor platform was cut out of the frame using a 5-inch cutting disk. Similarly, the slot for the shaft was not wide enough for the dimensions of our designed shaft. So a wider slot was cut to accommodate this requirement.

Apart from that, we had to weld an additional C-shaped channel to mount one of the piezoelectric sensors. Hence, the required length of beam was bought from the market and welded to the right position on the frame, as indicated in the CAD model as well.

Base Plate:

For the required assembly operations, we needed to make the following modifications to the plate:

- Four rectangular slots to assemble the motor onto the frame. These were done using *oxy-acetylene torch* cutting.
- Mild steel mountings to be welded on the base plate for assembling the shaft onto the base plate.
- Five holes in the base plate to assemble it on the frame. These were drilled using *Magnetic Drill Press*.

The motor is assembled to the plate using an M8 bolt, while the plate itself is assembled to the frame using M13 bolts.

The shaft was premounted on the base plate. The motor was mounted on its slots, and the V-belt was finally mounted on the motor side pulley and the shaft pulley while keeping the center distance, such that there was no sag in the belt.

Signal Conditioning and Instrumentation:

The signal conditioning unit is an integral part of the data acquisition and processing unit of our machine, the main purpose of which is to convert the data received from the piezoelectric sensors to volts. The main function of a signal conditioning unit includes signal conversion, linearization, amplification, and filtering.

A signal conditioner is a device that functions as a bridge between the sensor and the data acquisition system, modifying the raw analogue output signals that sensors produce. Through these changes, the various signals are transformed into signals that are compatible with devices used for process monitoring and control.

Its main function is to process and prepare signals from sensors or other sources so that downstream parts like microcontrollers or analog-to-digital converters (ADCs) can process them accurately and dependably.

Following are the features of the signal conditioning unit that are of key importance:

1. **Amplification:** One of the most common functions of a signal conditioning unit is to amplify weak signals to levels suitable for further processing. This is especially

important in sensor applications where the output signal may be small and needs to be boosted for accurate measurement. Operational amplifiers (op-amps) are commonly used for this purpose, providing adjustable gain to scale the input signal.

2. **Filtering:** Signal conditioning units often include filtering circuits to remove unwanted noise or interference from the signal. Filters can be implemented using passive components such as resistors, capacitors, and inductors, or active components like op-amps configured as filter stages. Common types of filters include low-pass, high-pass, band-pass, and notch filters, each tailored to the specific frequency characteristics of the noise or interference being targeted.
3. **Voltage/Current Conversion:** Signal conditioning units may also perform voltage-to-current or current-to-voltage conversion to interface between different types of sensors and measurement systems. This conversion ensures compatibility between the signal source and the input requirements of downstream components.

Following are the circuits that were employed, and a PCB circuit was made with a custom Power supply.

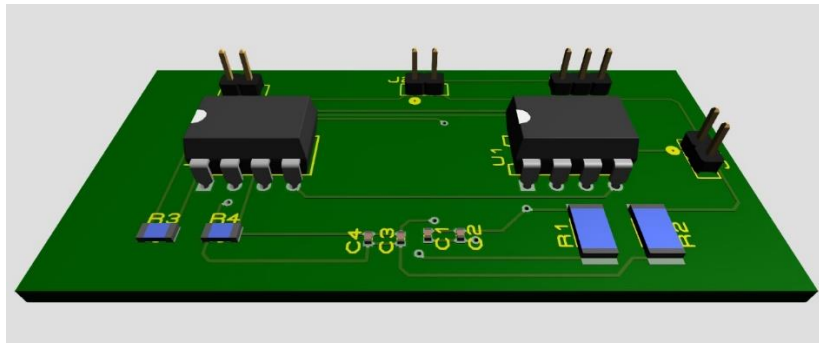


Figure 24



Figure 25

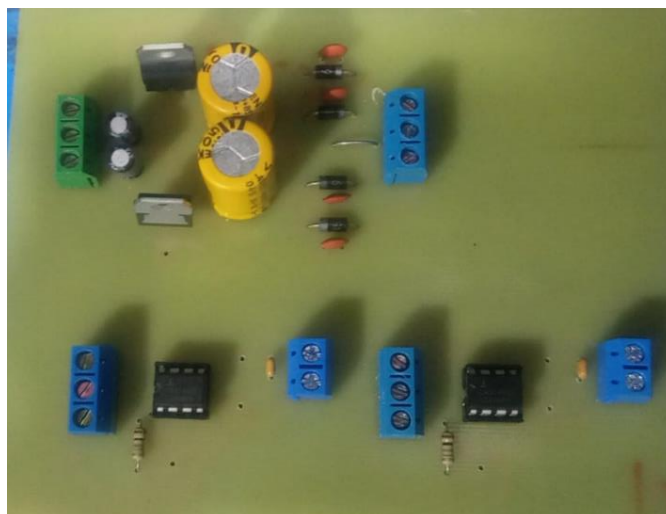


Figure 26

The STM32 card is used for the data acquisition in the machine. The STM32F407 is a microcontroller produced by STMicroelectronics, belonging to the STM32 family. It features a high-performance ARM Cortex-M4 core with floating-point unit (FPU), which makes it suitable for a wide range of applications including data acquisition.

To create a data acquisition card using the STM32F407, you would typically integrate this microcontroller onto a PCB along with the necessary peripheral components such as analog-to-digital converters (ADCs), digital-to-analog converters (DACs), communication interfaces (e.g., UART, SPI, I2C)

The key features of STM32 that make it suitable for data acquisition include the following:

1. **High-Performance ARM Cortex-M4 Core:** The Cortex-M4 core operates at up to 168 MHz, providing high processing power for real-time data processing tasks.
2. **Memory:** The STM32F407 typically comes with various memory options including Flash memory for program storage and RAM for data storage.
3. **Analog-to-Digital Converters (ADCs):** The STM32F407 features multiple ADC channels with resolutions up to 12 bits, enabling it to convert analog signals into digital data for processing.
4. **Communication Interfaces:** It includes multiple communication interfaces such as UART, SPI, I2C, USB, and Ethernet, which can be used to interface with external devices or communicate with a host computer.

5. **Timers and PWM Outputs:** Timers and PWM outputs can be utilized for generating precise timing signals or controlling external devices.
6. **GPIOs:** General-purpose input/output pins allow interfacing with external sensors, actuators, or other peripheral devices.
7. **Integrated Development Environment (IDE) Support:** STMicroelectronics provides the STM32CubeIDE, a free development environment for STM32 microcontrollers, which simplifies the development process.

Analog to Digital Conversion using STM32:

STM32 Microcontrollers have multiple ADC units which allow simultaneous sampling. The ADC in STM32 microcontrollers offer resolutions ranging from 6 to 16 bits. The sampling rate of the ADC determines how frequently it samples the input analog signal. The controller supports different conversion modes which include single conversion, continuous conversion, and scan mode. ADC conversions can be triggered by various events which include software triggers or external signals.

For configuration of STM32, we will have to do peripheral initialization and channel configuration. We must initialize the ADC peripheral by enabling its clock and configuring its operating parameters such as resolution, sampling rates and conversion mode.

The following are the functions that were used in STM32.

```
HAL_StatusTypeDef HAL_ADC_Start(ADC_HandleTypeDef *hadc);
```

```

HAL_StatusTypeDef HAL_ADC_Stop(ADC_HandleTypeDef *hadc);

HAL_StatusTypeDef HAL_ADC_PollForConversion(ADC_HandleTypeDef *hadc,
uint32_t Timeout);

uint32_t HAL_ADC_GetValue(ADC_HandleTypeDef *hadc);

void HAL_GPIO_EXTI_Callback(uint16_t GPIO_Pin);

HAL_GPIO_ReadPin(GPIO_TypeDef* GPIOx, uint16_t GPIO_Pin);

void HAL_GPIO_WritePin(GPIO_TypeDef* GPIOx, uint16_t GPIO_Pin, GPIO_PinState
PinState);

void HAL_GPIO_TogglePin(GPIO_TypeDef* GPIOx, uint16_t GPIO_Pin);

```

The Purpose of each of the following functions is as follows:

1. **HAL_StatusTypeDef HAL_ADC_Start(ADC_HandleTypeDef hadc);*
 - This function is used to start the ADC conversion process.
 - Parameters:
 - **hadc**: Pointer to the ADC handle structure.
 - Return Value:
 - HAL_OK if the ADC is successfully started.
 - Other HAL_StatusTypeDef values indicating an error.

2. **HAL_StatusTypeDef HAL_ADC_Stop(ADC_HandleTypeDef hadc);*

- This function is used to stop the ADC conversion process.
- Parameters:
 - **hadc**: Pointer to the ADC handle structure.
- Return Value:
 - HAL_OK if the ADC is successfully stopped.
 - Other HAL_StatusTypeDef values indicating an error.

3. **HAL_StatusTypeDef HAL_ADC_PollForConversion(ADC_HandleTypeDef hadc, uint32_t Timeout);*

- This function is used to poll for the end of ADC conversion.
- Parameters:
 - **hadc**: Pointer to the ADC handle structure.
 - **Timeout**: Timeout value in milliseconds.
- Return Value:
 - HAL_OK if the conversion is completed within the specified timeout period.
 - Other HAL_StatusTypeDef values indicating an error.

4. **uint32_t HAL_ADC_GetValue(ADC_HandleTypeDef hadc);*

- This function is used to retrieve the result of the last ADC conversion.
- Parameters:
 - **hadc**: Pointer to the ADC handle structure.
- Return Value:
 - The converted digital value of the last conversion.

5. **void HAL_GPIO_EXTI_Callback(uint16_t GPIO_Pin);**

- This function is a callback invoked when an external interrupt event occurs on a GPIO pin.
- Parameters:
 - **GPIO_Pin**: The pin number that triggered the interrupt.

6. *HAL_GPIO_ReadPin(GPIO_TypeDef GPIOx, uint16_t GPIO_Pin);**

- This function is used to read the input value of a specific GPIO pin.
- Parameters:
 - **GPIOx**: Pointer to the GPIO port (e.g., GPIOA, GPIOB, etc.).
 - **GPIO_Pin**: The pin number to read.
- Return Value:
 - GPIO_PIN_RESET or GPIO_PIN_SET, indicating the pin state.

7. *void HAL_GPIO_WritePin(GPIO_TypeDef GPIOx, uint16_t GPIO_Pin, GPIO_PinState PinState);**

- This function is used to set the output value of a specific GPIO pin.
- Parameters:
 - **GPIOx**: Pointer to the GPIO port (e.g., GPIOA, GPIOB, etc.).
 - **GPIO_Pin**: The pin number to set.
 - **PinState**: The desired state of the pin (GPIO_PIN_RESET or GPIO_PIN_SET).

8. *void HAL_GPIO_TogglePin(GPIO_TypeDef GPIOx, uint16_t GPIO_Pin);**

- This function is used to toggle the output value of a specific GPIO pin.
- Parameters:
 - **GPIOx**: Pointer to the GPIO port (e.g., GPIOA, GPIOB, etc.).
 - **GPIO_Pin**: The pin number to toggle.

Following is the pin layout of the STM32 board. The pins that were engaged are shown highlighted in the layout below and each of the pin functions are described as above.

Application Development:

The android application was developed in the MIT App Inventor platform. The application is connected to the DAQ via an HC-05 Bluetooth module.

The operator can easily transmit the values like rim diameter, distance from the machine, etc simply by typing these values in the UI of the application.

After the motor runs for 8 seconds, the magnitude of imbalance is indicated on the mobile screen.

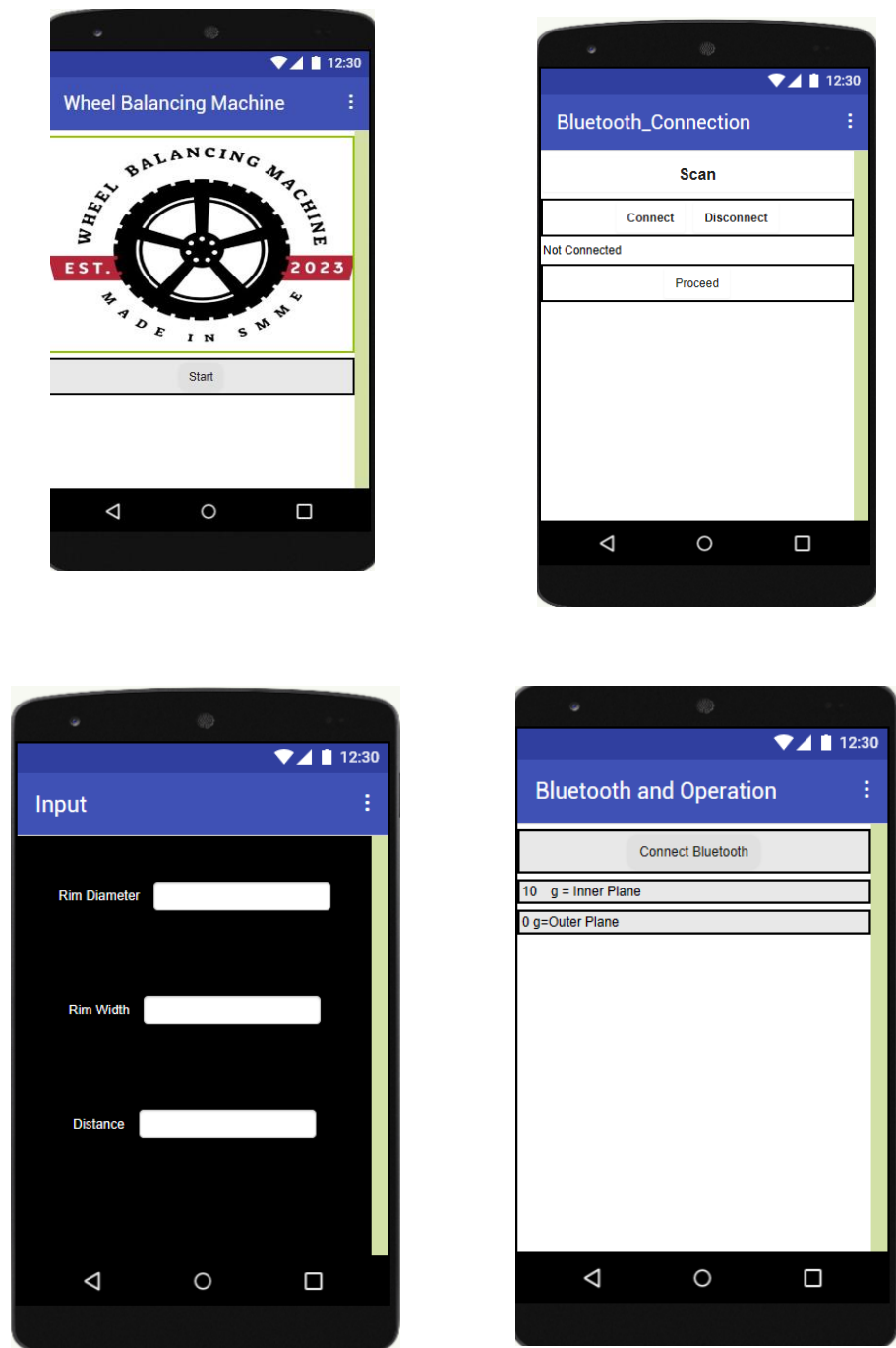


Figure 27: Mobile Application

Testing of Components:

1. Motor:

The motor was the first component that was tested. To ensure that no damage occurs to the induction wires, the motor was connected to a circuit breaker and was not plugged in directly to power. In addition to a circuit breaker, a regulator was added to control the speed/rpm of the motor. This completed the basic circuit of the motor. To control the timing of the motor, an additional microcontroller was used. As commercially available wheel balancing machines usually employ a machine run time of 8 seconds, a code was written that would run the motor for 8 seconds, through a relay module which was as follows:



Figure 28: Relay Module

The microcontroller code was as follows:

```
int mot_relay=8;

void setup() {
  pinMode(mot_relay,OUTPUT);
  digitalWrite(mot_relay,LOW);
  Serial.begin(9600); // opens serial port, sets data rate to 9600 bps
}

void loop() {

  delay(5000);
  digitalWrite(mot_relay,HIGH);
  delay(10000);
  digitalWrite(mot_relay,LOW);
  delay(2000);
}
```

NOTE: To run the machine, we reset the Arduino board while the voltage regulator is set to zero position. Gradually we rotate the position of the voltage regulator to allow the motor to run smoothly. To ensure the safety of the equipment, we reset the voltage regulator to zero every time the cycle of the machine was run.

2. Piezoelectric sensors signal:

The piezoelectric signals were tested independently before they were mounted in the balancing machine with different weights applied. The voltage range observed for the circuit using the digital multimeter 0.5 to 3V. The signal observed were as follows:

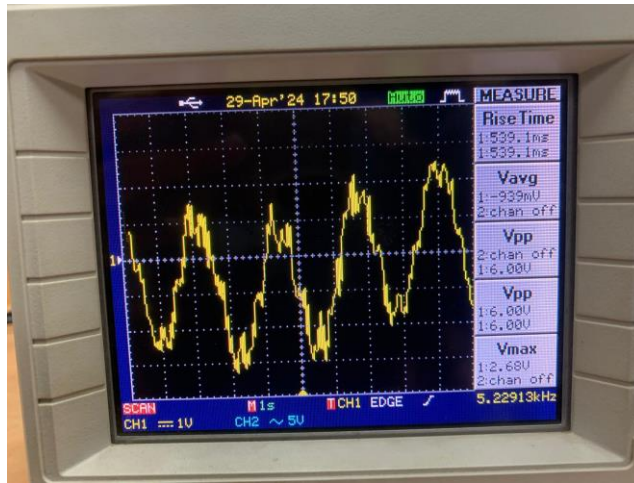


Figure 29

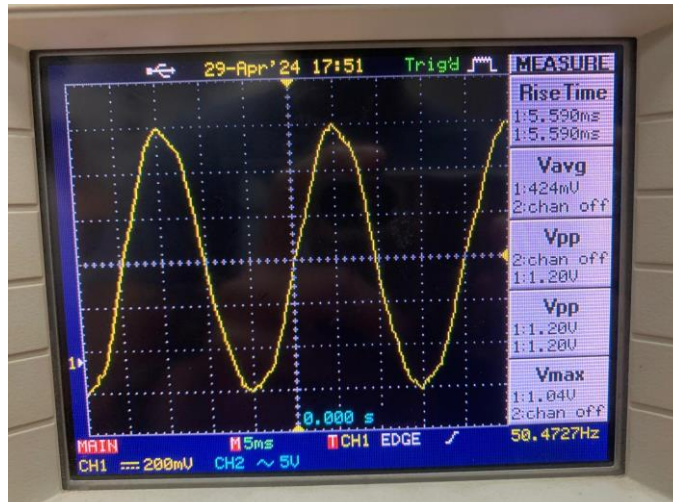


Figure 30

NOTE: Initially we obtained a sinusoidal signal. But the correct signal was obtained after we set voltage divisions and time divisions in the oscilloscope.


```

* This software is licensed under terms that can be found in the LICENSE file
* in the root directory of this software component.
* If no LICENSE file comes with this software, it is provided AS-IS.
*

*****

*/
/* USER CODE END Header */
/* Includes -----
*/
#include "main.h"

/* Private includes -----
*/
/* USER CODE BEGIN Includes */

/* USER CODE END Includes */

/* Private typedef -----
*/
/* USER CODE BEGIN PTD */

/* USER CODE END PTD */

/* Private define -----
*/
/* USER CODE BEGIN PD */

/* USER CODE END PD */

/* Private macro -----
*/
/* USER CODE BEGIN PM */

/* USER CODE END PM */

```

```

/* Private variables -----
*/
ADC_HandleTypeDef hadc1;

/* USER CODE BEGIN PV */

/* USER CODE END PV */

/* Private function prototypes -----
*/
void SystemClock_Config(void);
static void MX_GPIO_Init(void);
static void MX_ADC1_Init(void);
/* USER CODE BEGIN PFP */

/* USER CODE END PFP */

/* Private user code -----
*/
/* USER CODE BEGIN 0 */
void ADC_Select_CH2 (void)
{
    ADC_ChannelConfTypeDef sConfig = {0};
    sConfig.Channel = ADC_CHANNEL_2;
    sConfig.Rank = 1;
    sConfig.SamplingTime = ADC_SAMPLETIME_480CYCLES;
    if (HAL_ADC_ConfigChannel(&hadc1, &sConfig) != HAL_OK)
    {
        Error_Handler();
    }
}

__IO uint16_t Valarray1[1000];
__IO uint16_t Valarray2[1000];

```

```

uint16_t w=1;
/* USER CODE END 0 */

/**
 * @brief The application entry point.
 * @retval int
 */
int main(void)
{

    /* USER CODE BEGIN 1 */

    /* USER CODE END 1 */

    /* MCU Configuration-----*/

    /* Reset of all peripherals, Initializes the Flash interface and the Systick.
    */
    HAL_Init();

    /* USER CODE BEGIN Init */

    /* USER CODE END Init */

    /* Configure the system clock */
    SystemClock_Config();

    /* USER CODE BEGIN SysInit */

    /* USER CODE END SysInit */

    /* Initialize all configured peripherals */
    MX_GPIO_Init();

```



```

MX_ADC1_Init();
/* USER CODE BEGIN 2 */

/* USER CODE END 2 */

/* Infinite loop */
/* USER CODE BEGIN WHILE */
while (1)
{
    /* USER CODE END WHILE */

    /* USER CODE BEGIN 3 */

}
/* USER CODE END 3 */
}

/**
 * @brief System Clock Configuration
 * @retval None
 */
void SystemClock_Config(void)
{
    RCC_OscInitTypeDef RCC_OscInitStruct = {0};
    RCC_ClkInitTypeDef RCC_ClkInitStruct = {0};

    /** Configure the main internal regulator output voltage
    */
    __HAL_RCC_PWR_CLK_ENABLE();
    __HAL_PWR_VOLTAGESCALING_CONFIG(PWR_REGULATOR_VOLTAGE_SCALE1);

    /** Initializes the RCC Oscillators according to the specified parameters
    * in the RCC_OscInitTypeDef structure.
    */

```

```

RCC_OscInitStruct.OscillatorType = RCC_OSCILLATORTYPE_HSI;
RCC_OscInitStruct.HSIState = RCC_HSI_ON;
RCC_OscInitStruct.HSICalibrationValue = RCC_HSICALIBRATION_DEFAULT;
RCC_OscInitStruct.PLL.PLLState = RCC_PLL_ON;
RCC_OscInitStruct.PLL.PLLSource = RCC_PLLSOURCE_HSI;
RCC_OscInitStruct.PLL.PLLM = 8;
RCC_OscInitStruct.PLL.PLLN = 168;
RCC_OscInitStruct.PLL.PLLP = RCC_PLLP_DIV2;
RCC_OscInitStruct.PLL.PLLQ = 4;
if (HAL_RCC_OscConfig(&RCC_OscInitStruct) != HAL_OK)
{
    Error_Handler();
}

/** Initializes the CPU, AHB and APB buses clocks
 */
RCC_ClkInitStruct.ClockType = RCC_CLOCKTYPE_HCLK|RCC_CLOCKTYPE_SYSCLK
                               |RCC_CLOCKTYPE_PCLK1|RCC_CLOCKTYPE_PCLK2;
RCC_ClkInitStruct.SYSCLKSource = RCC_SYSCLKSOURCE_PLLCLK;
RCC_ClkInitStruct.AHBCLKDivider = RCC_SYSCLK_DIV1;
RCC_ClkInitStruct.APB1CLKDivider = RCC_HCLK_DIV4;
RCC_ClkInitStruct.APB2CLKDivider = RCC_HCLK_DIV2;

if (HAL_RCC_ClockConfig(&RCC_ClkInitStruct, FLASH_LATENCY_5) != HAL_OK)
{
    Error_Handler();
}
}

/**
 * @brief ADC1 Initialization Function
 * @param None
 * @retval None
 */

```

```

static void MX_ADC1_Init(void)
{

    /* USER CODE BEGIN ADC1_Init 0 */

    /* USER CODE END ADC1_Init 0 */

    ADC_ChannelConfTypeDef sConfig = {0};

    /* USER CODE BEGIN ADC1_Init 1 */

    /* USER CODE END ADC1_Init 1 */

    /** Configure the global features of the ADC (Clock, Resolution, Data Alignment
    and number of conversion)
    */
    hadc1.Instance = ADC1;
    hadc1.Init.ClockPrescaler = ADC_CLOCK_SYNC_PCLK_DIV2;
    hadc1.Init.Resolution = ADC_RESOLUTION_12B;
    hadc1.Init.ScanConvMode = ENABLE;
    hadc1.Init.ContinuousConvMode = ENABLE;
    hadc1.Init.DiscontinuousConvMode = DISABLE;
    hadc1.Init.ExternalTrigConvEdge = ADC_EXTERNALTRIGCONVEDGE_NONE;
    hadc1.Init.ExternalTrigConv = ADC_SOFTWARE_START;
    hadc1.Init.DataAlign = ADC_DATAALIGN_RIGHT;
    hadc1.Init.NbrOfConversion = 1;
    hadc1.Init.DMAContinuousRequests = DISABLE;
    hadc1.Init.EOCSelection = ADC_EOC_SEQ_CONV;
    if (HAL_ADC_Init(&hadc1) != HAL_OK)
    {
        Error_Handler();
    }
}

```

```

    /** Configure for the selected ADC regular channel its corresponding rank in
    the sequencer and its sample time.
    */
    sConfig.Channel = ADC_CHANNEL_2;
    sConfig.Rank = 1;
    sConfig.SamplingTime = ADC_SAMPLETIME_3CYCLES;
    if (HAL_ADC_ConfigChannel(&hadc1, &sConfig) != HAL_OK)
    {
        Error_Handler();
    }
    /* USER CODE BEGIN ADC1_Init 2 */

    /* USER CODE END ADC1_Init 2 */

}

/**
 * @brief GPIO Initialization Function
 * @param None
 * @retval None
 */
static void MX_GPIO_Init(void)
{
    GPIO_InitTypeDef GPIO_InitStruct = {0};
    /* USER CODE BEGIN MX_GPIO_Init_1 */
    /* USER CODE END MX_GPIO_Init_1 */

    /* GPIO Ports Clock Enable */
    __HAL_RCC_GPIOA_CLK_ENABLE();
    __HAL_RCC_GPIOD_CLK_ENABLE();

    /*Configure GPIO pin Output Level */
    HAL_GPIO_WritePin(GPIOD, GPIO_PIN_12|GPIO_PIN_13|GPIO_PIN_14|GPIO_PIN_15,
GPIO_PIN_RESET);

```

```

/*Configure GPIO pin : PA0 */
GPIO_InitStruct.Pin = GPIO_PIN_0;
GPIO_InitStruct.Mode = GPIO_MODE_IT_RISING;
GPIO_InitStruct.Pull = GPIO_PULLDOWN;
HAL_GPIO_Init(GPIOA, &GPIO_InitStruct);

/*Configure GPIO pins : PD12 PD13 PD14 PD15 */
GPIO_InitStruct.Pin = GPIO_PIN_12|GPIO_PIN_13|GPIO_PIN_14|GPIO_PIN_15;
GPIO_InitStruct.Mode = GPIO_MODE_OUTPUT_PP;
GPIO_InitStruct.Pull = GPIO_NOPULL;
GPIO_InitStruct.Speed = GPIO_SPEED_FREQ_LOW;
HAL_GPIO_Init(GPIOD, &GPIO_InitStruct);

/* EXTI interrupt init*/
HAL_NVIC_SetPriority(EXTI0_IRQn, 0, 0);
HAL_NVIC_EnableIRQ(EXTI0_IRQn);

/* USER CODE BEGIN MX_GPIO_Init_2 */
/* USER CODE END MX_GPIO_Init_2 */
}

/* USER CODE BEGIN 4 */
void HAL_GPIO_EXTI_Callback(uint16_t GPIO_Pin)
{
    UNUSED(GPIO_Pin);
    HAL_GPIO_TogglePin(GPIOD, GPIO_PIN_12);
    ADC_Select_CH2();

    HAL_ADC_Start(&hadc1);
    HAL_ADC_PollForConversion(&hadc1, HAL_MAX_DELAY);
    Valarray1[w] = HAL_ADC_GetValue(&hadc1);
    Valarray2[w] = Valarray2[w-1] + 11.25;
    HAL_ADC_Stop(&hadc1);
}

```

```

        w++;

    }

    /* USER CODE END 4 */

    /**
     * @brief This function is executed in case of error occurrence.
     * @retval None
     */
    void Error_Handler(void)
    {
        /* USER CODE BEGIN Error_Handler_Debug */
        /* User can add his own implementation to report the HAL error return state */
        __disable_irq();
        while (1)
        {
        }
        /* USER CODE END Error_Handler_Debug */
    }

    #ifdef USE_FULL_ASSERT
    /**
     * @brief Reports the name of the source file and the source line number
     * where the assert_param error has occurred.
     * @param file: pointer to the source file name
     * @param line: assert_param error line source number
     * @retval None
     */
    void assert_failed(uint8_t *file, uint32_t line)
    {
        /* USER CODE BEGIN 6 */
        /* User can add his own implementation to report the file name and line number,

```

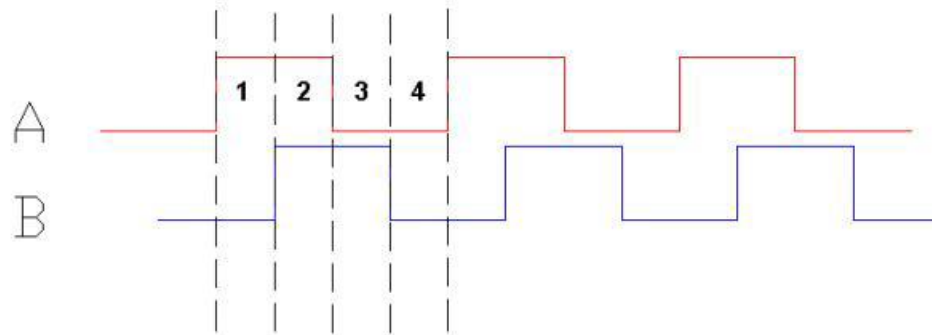
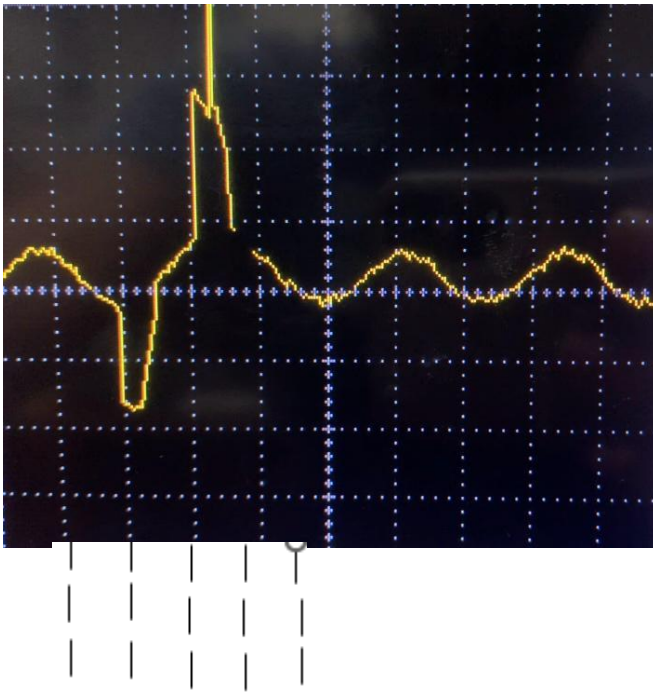
```

    ex: printf("Wrong parameters value: file %s on line %d\r\n", file, line)
*/
/* USER CODE END 6 */
}
#endif /* USE_FULL_ASSERT */

```

The voltages read from the piezoelectric sensor and the respective positions were displayed as follows:

Valarray2	volatile uint16_t [1000]	[1000]	0x20000844
[0..99]		[1000]	
Valarray2[0]	volatile uint16_t	0	0x20000844
Valarray2[1]	volatile uint16_t	11	0x20000846
Valarray2[2]	volatile uint16_t	22	0x20000848
Valarray2[3]	volatile uint16_t	33	0x2000084a
Valarray2[4]	volatile uint16_t	44	0x2000084c
Valarray2[5]	volatile uint16_t	55	0x2000084e
Valarray2[6]	volatile uint16_t	66	0x20000850
Valarray2[7]	volatile uint16_t	77	0x20000852
Valarray2[8]	volatile uint16_t	88	0x20000854
Valarray2[9]	volatile uint16_t	99	0x20000856
Valarray2[10]	volatile uint16_t	110	0x20000858
Valarray2[11]	volatile uint16_t	121	0x2000085a
Valarray2[12]	volatile uint16_t	132	0x2000085c
Valarray2[13]	volatile uint16_t	143	0x2000085e
Valarray2[14]	volatile uint16_t	154	0x20000860
Valarray2[15]	volatile uint16_t	165	0x20000862
Valarray2[16]	volatile uint16_t	176	0x20000864
Valarray2[17]	volatile uint16_t	187	0x20000866
Valarray2[18]	volatile uint16_t	198	0x20000868
Valarray2[19]	volatile uint16_t	209	0x2000086a
Valarray2[20]	volatile uint16_t	220	0x2000086c
Valarray2[21]	volatile uint16_t	231	0x2000086e
Valarray2[22]	volatile uint16_t	242	0x20000870
Valarray2[23]	volatile uint16_t	253	0x20000872
Valarray2[24]	volatile uint16_t	264	0x20000874
Valarray2[25]	volatile uint16_t	275	0x20000876
Valarray2[26]	volatile uint16_t	286	0x20000878



As mentioned before, only the rising edge of the optical encoder signal will be used to record the voltages obtained from the piezoelectric sensor through the signal conditioning circuit.

Valarray2	volatile uint16_t [1000]	[1000]	0x20000844
[0..99]		[1000]	
Valarray2[0]	volatile uint16_t	0	0x20000844
Valarray2[1]	volatile uint16_t	11	0x20000846
Valarray2[2]	volatile uint16_t	22	0x20000848
Valarray2[3]	volatile uint16_t	33	0x2000084a
Valarray2[4]	volatile uint16_t	44	0x2000084c
Valarray2[5]	volatile uint16_t	55	0x2000084e
Valarray2[6]	volatile uint16_t	66	0x20000850
Valarray2[7]	volatile uint16_t	77	0x20000852
Valarray2[8]	volatile uint16_t	88	0x20000854
Valarray2[9]	volatile uint16_t	99	0x20000856
Valarray2[10]	volatile uint16_t	110	0x20000858
Valarray2[11]	volatile uint16_t	121	0x2000085a
Valarray2[12]	volatile uint16_t	132	0x2000085c
Valarray2[13]	volatile uint16_t	143	0x2000085e
Valarray2[14]	volatile uint16_t	154	0x20000860
Valarray2[15]	volatile uint16_t	165	0x20000862
Valarray2[16]	volatile uint16_t	176	0x20000864
Valarray2[17]	volatile uint16_t	187	0x20000866
Valarray2[18]	volatile uint16_t	198	0x20000868
Valarray2[19]	volatile uint16_t	209	0x2000086a
Valarray2[20]	volatile uint16_t	220	0x2000086c
Valarray2[21]	volatile uint16_t	231	0x2000086e
Valarray2[22]	volatile uint16_t	242	0x20000870
Valarray2[23]	volatile uint16_t	253	0x20000872
Valarray2[24]	volatile uint16_t	264	0x20000874
Valarray2[25]	volatile uint16_t	275	0x20000876
Valarray2[26]	volatile uint16_t	286	0x20000878

IMPORTANT NOTE: One of the very major Mistakes made were that we did not use a common ground between the STM32 data acquisition card and conditioning unit.

4. Rotary Encoder:

The rotary encoder consists of important components such as the *zero index* which resets at every starting point of the shaft.

Moreover, a trigger pulse is also used to mark the completion of one revolution.

Two photo-interrupters record the angular movement in both the anticlockwise and clockwise direction of shaft.

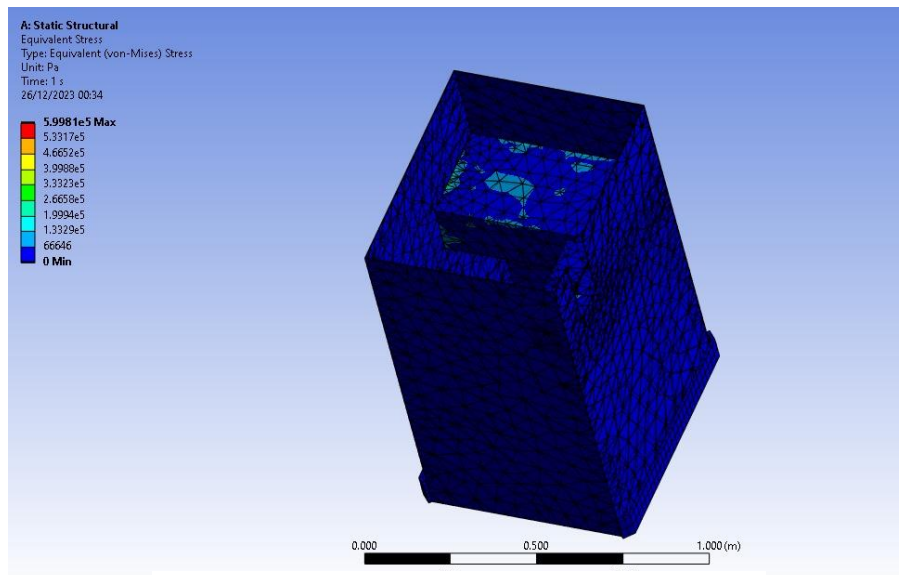
CHAPTER 4: RESULTS AND DISCUSSIONS

The outcome of this project is to design and manufacture a cost-effective wheel balancing machine with modular circuitry while not compromising on its accuracy. We implemented the methodology as stated in Chapter 3 of the report and were successful in developing a design for the machine.

Before moving on to the manufacturing part, we had to make sure that our design was flawless and would not break down upon operating the machine. For this purpose, we conducted the finite element and modal analyses on the design. The following sections discuss the analyses as conducted in Ansys:

Analysis of the Outer Frame of the Machine:

During the design phase, we came up with two different designs for the outer frame of the machine. To choose amongst the two, we conducted a finite element analysis which is shown in the images below:



Design 1

Figure 32: Von-Mises stress

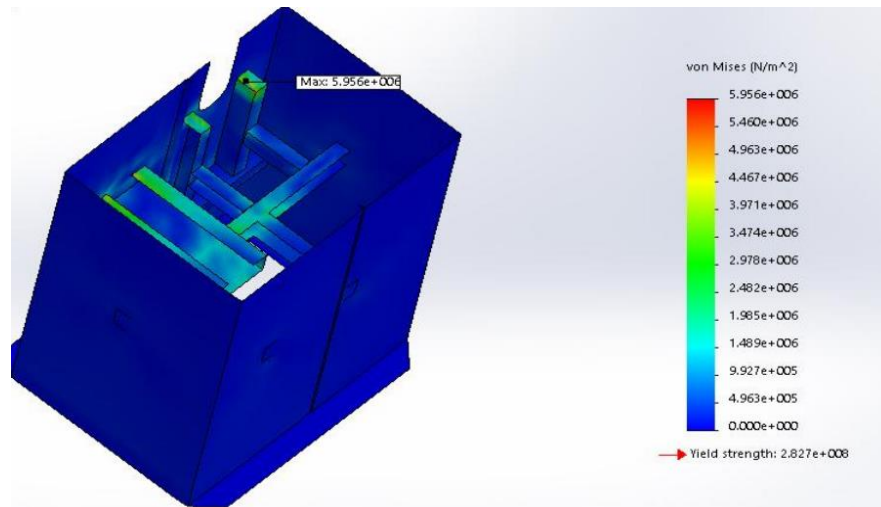


Figure 33: 2nd Iteration

We can see from the figure of Design 2 that there are higher values of stresses being generated on the points where the motor and shaft would be mounted. On the other hand, design 1 has low stresses in the frame's geometry. Based on this strong reason, we finalized the design 1 to be the design of our machine.

Analyses of the Shaft:

The shaft is the most important part of the machine on the wheel which is to be balanced and will be mounted. After designing the shaft (discussed in Chapter 3), it was necessary to conduct the analyses to verify that the design was safe for operation.

We conducted two types of analyses for the shaft:

1. Analysis with static load only (the weight of the wheel when the machine was stationary).

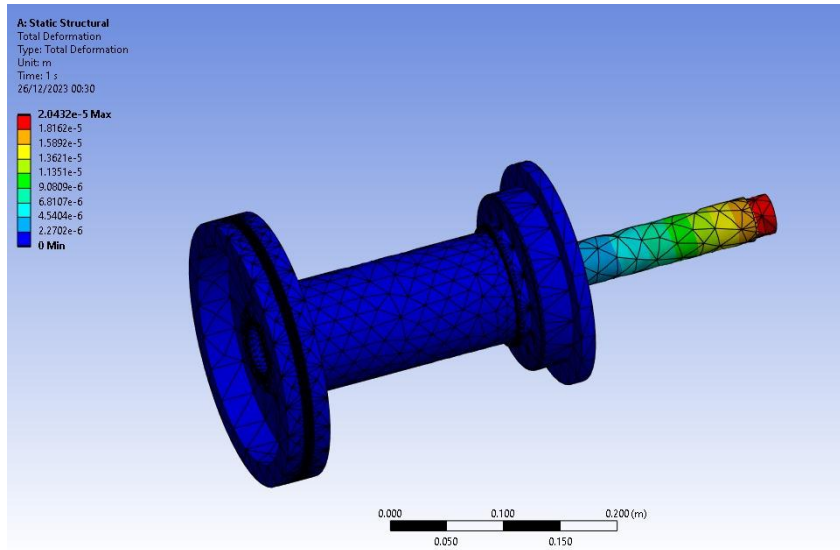


Figure 34: Total Deformation

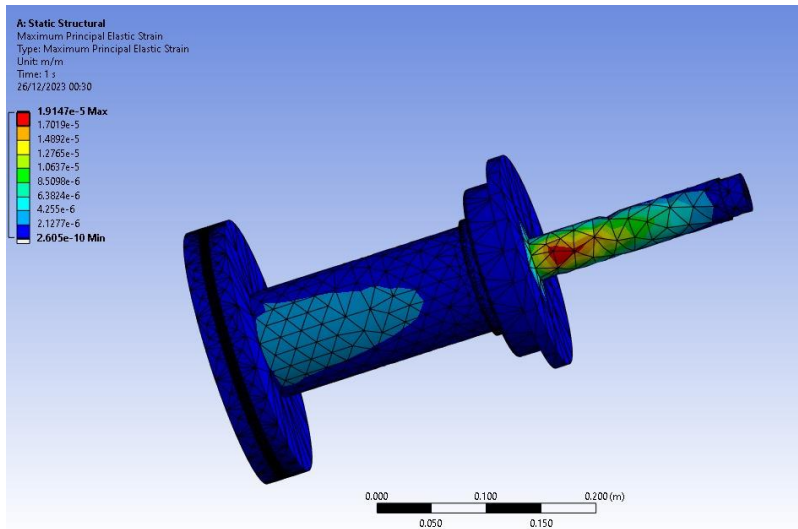


Figure 35: Stress

2. Analysis with static load as well as rotating load during the machine's operation.

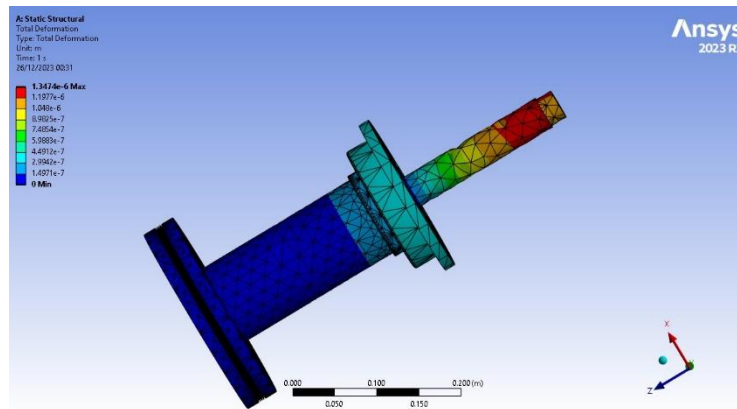


Figure 36: Total Deformation

Modal Analysis:

The natural frequency analysis conducted in the modal solver of Ansys shows us 5 different natural frequencies ranging from approximately 82 Hz to 205 Hz, whereas the operating frequency of our machine is 16.667 Hz which is very far from any of the natural frequencies. This shows that the maximum frequency ratio is 0.203, as compared to 1 where resonance occurs.

The results of the analysis are shown in the below images:

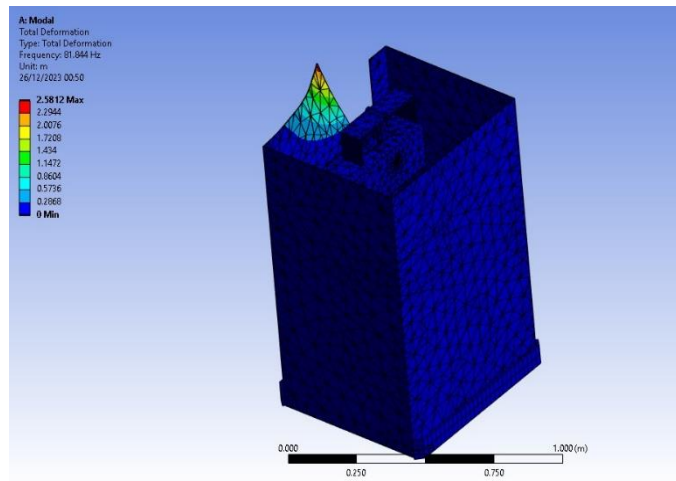


Figure 37: Mode 1: 81.88 Hz

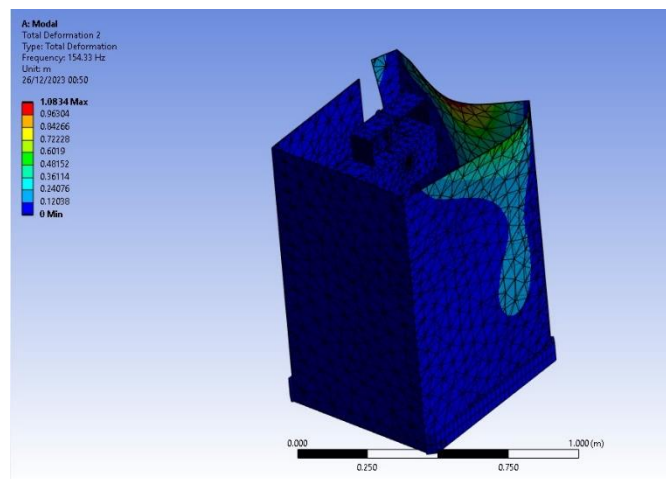


Figure 38: Mode 2: 154.33 Hz

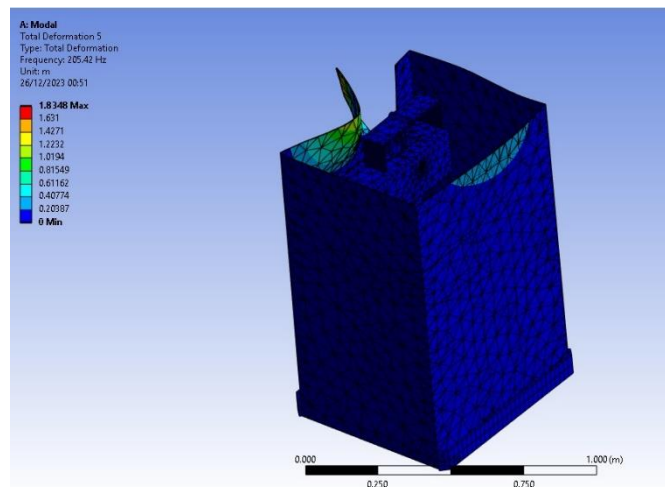


Figure 39: Mode 3: 175 Hz

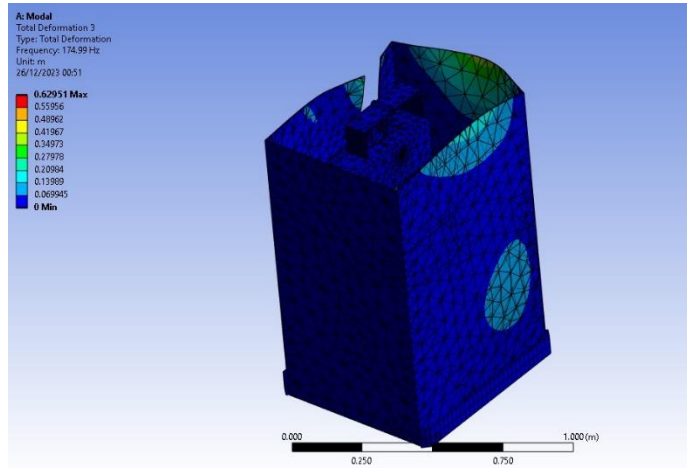


Figure 40 Mode 4: 188 Hz

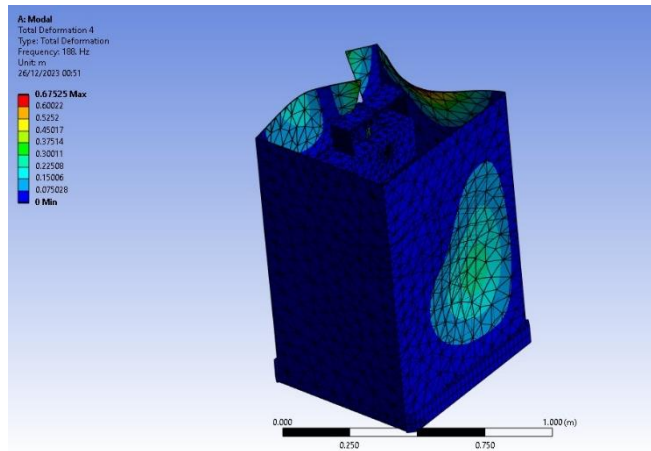


Figure 41: Mode 5: 205.42 Hz

CHAPTER 5: CONCLUSION AND RECOMMENDATION

Concluding Remarks:

In the culmination of our final year design project, we successfully navigated through the intricate process of conceptualizing, designing, and analyzing a dynamic wheel balancing machine. This endeavor was driven by the necessity to address the prevalent challenges faced in conventional wheel balancing techniques, ultimately aiming to enhance efficiency and accuracy in automotive maintenance procedures. Our journey commenced with a comprehensive understanding of the abstract concept, followed by a detailed breakdown of the components and systems essential for the functionality of the machine. Central to our design is the drive shaft assembly, crafted to ensure optimal performance and durability. This critical component serves as the backbone of the machine, facilitating the dynamic balancing process with precision and efficiency. Our selection of materials and engineering principles ensured that the drive shaft assembly met rigorous standards of reliability and performance, thereby laying a robust foundation for the overall functionality of the machine. Moreover, our design process involved careful consideration of various elements, including frame construction, motor selection, pulley and belt drive systems, and the integration of piezo sensors for real-time data acquisition. Each component was carefully scrutinized, with emphasis placed on achieving synergy and coherence within the system. Through rigorous analysis and iterative refinement, we have successfully laid the basis for the development of a dynamic wheel balancing machine as well as a mobile application that not only meets but exceeds industry standards in terms of accuracy, efficiency, and reliability. This project stands as a testament to our collective dedication, ingenuity, and perseverance in the pursuit of engineering excellence. As we conclude this chapter, we look forward to the continued evolution and real-world application of our innovative solution in automotive maintenance practices.

Future Recommendations:

Future recommendations include refinement of the data acquisition system to appropriate accuracy and integration with an advanced mobile app having a better GUI. The shaft assembly can be upgraded to accommodate different tire sizes and range of automobiles. Furthermore, the machine can have advanced features that include automated calibration and use of laser to measure position.

REFERENCES

- [1] A. M. Farooq, A. Rafaeh, H. Naseem and H. Iftikhar, "DESIGN AND MANUFACTURING OF WHEEL BALANCING MACHINE," School of Mechanical & Manufacturing Engineering, NUST, ISLAMABAD, PAKISTAN, 2018.
- [2] "Car and Truck Wheel Balancing Machine," [Online]. Available: <https://risingsuntools.en.made-in-china.com/product/XBLxSCFHyDRO/China-Car-and-Truck-Wheel-Balancing-Machine.html>.
- [3] "Electronic Wheel Balancing Machine B 335 EVO," [Online]. Available: <https://gietsl.com/product/equilibradora-de-ruedas-electronica-b-335-evo/>.
- [4] P. R. Salave, A. K. Chouhan, S. K. Mane, J. D. Bagate, G. B. Patil and S. T. Dudhbhate, "Design and Development of Wheel Balancing," Department of Mechanical Engineering, Savitribai Phule Pune University, Pune, Maharashtra, 2019.
- [5] Gates, "Industrial V-Belt Design Manual," Gates, 2012.
- [6] E. Yilmaz, "Microcomputer Based Instrumentation for Student Designed Wheel Balancing Machine," Department of Technology, University of Maryland Eastern Shore, Princess Anne, MD 21853, 1998.
- [7] Atlas, "Atlas WB49-2 Pro Self-Calibrating Computer Wheel Balancer," Atlas Equipment.

- [8] A. Devices, "CN0350 Data Acquisition," Analogue Device, Inc.. [Online]. Available: <https://www.analog.com/en/design-center/reference-designs/circuits-from-the-lab/CN0350.html#rd-overview>.
- [9] S. Life, "STM32Fxxx Discovery Board Catalogue," August 2020. [Online]. Available: <https://www.st.com/resource/en/datasheet/stm32f405rg.pdf>.
- [10] A. Nabar, "A STUDY OF OPERATIONAL VARIABLES INFLUENCING WHEEL BALANCING MEASUREMENTS," The University of North Carolina, Charlotte, 2016.
- [11] Y. Machinery, "2D Automatic Measurement Wheel Balancing Machine," Yuan Machinery Professional Equipment, [Online]. Available: <https://ms.yuanmech.com/car-wheel-balancer/60936168.html>.
- [12] Tutorialspoint, "Reactnative Tutorial," Tutorialspoint, [Online]. Available: https://www.tutorialspoint.com/react_native/index.htm.

APPENDIX I: TITLE OF APPENDIX I

APPENDIX I: PROPERTY TABLES

Table 2: 1023 Carbon Steel

Mechanical Properties	Metric	Imperial
Tensile strength	425 MPa	61600 psi
Yield strength	360 MPa	52200 psi
Shear modulus	80.0 GPa	11600 ksi
Bulk modulus	140 GPa	20300 ksi
Elastic modulus	190-210 GPa	27557-30458 ksi
Poisson's ratio	0.27-0.30	0.27-0.30
Elongation at break	15%	15%
Reduction of area	40%	40%
Hardness Brinell	121	121
Hardness, Knoop	140	140
Hardness, Rockwell B	68	68
Hardness, Vickers	126	126
Machinability	65	65

Table 3: Mild Steel

Mechanical Properties	Metric	Imperial
Hardness, Brinell	126	126
Hardness, Knoop	145	145
Hardness, Rockwell B	71	71
Hardness, Vickers	131	131
Tensile Strength, Ultimate	440 MPa	63800 psi
Tensile Strength, Yield	370 MPa	53700 psi
Elongation at Break (In 50 mm)	15.0 %	15.0 %
Reduction of Area	40.0 %	40.0 %
Modulus of Elasticity	205 GPa	29700 ksi
Bulk Modulus (Typical for steel)	140 GPa	20300 ksi
Poisson's Ratio (Typical for Steel)	0.290	0.290
Machinability	70 %	70 %
Shear Modulus (Typical for steel)	80.0 GPa	11600 ksi