DEVELOPMENT OF A STRAIN GAUGE FOR MONITORING SYSTEM THAT CAN BE APPLIED TO WIND TURBINE BLADES STRESS TESTING IN THE LABORATORY

Thesis report is submitted to the School of Engineering and Information Technology Murdoch University, Perth as a main part of ENG 470 Engineering Honours Thesis for majors in Industrial Computer System and Renewable Energy Engineering

Student Name:
Mohd Shahrul Nizam AZMI

Supervisor:
Dr Gareth Lee
Dr Jonathan Whale

Unit Coordinator:
Professor Parisa Arabzadeh Bahri

2nd July 2018

Murdoch University © 2018
Author’s Declaration

I declare that the content of this thesis document is my own work and research which has not been submitted to any agency or tertiary education institution. I also state that this document has not been published yet for any requirement under all circumstances before.

_________________________

(Mohd Shahrul Nizam AZMI)
Abstract

This document will provide information about the approach that used to get the reading information from the wind blade in laboratory. This part is mainly focusing on getting a signal reading from a strain gauge to Arduino. The development of this project was based upon the requirement to study the effect of the turbulence that might affect the wind turbine blade. The investigation of the load that the blade experienced will then be analysed to ensure the wind turbine is able to maximise the production and also take into account the safety issues that arise according to the standard after project design constructing and implementation occurred in the real system. This document will show the details of the instruments and the design approach that enabled capturing of the load data in a controlled environment. In field testing, strain gauges on the root of turbine blades capture the blade flapwise and edgewise loading. This project focuses on development of a monitoring system to capture strain gauge data. To simplify the experiment, strain gauge are loaded in the controlled environment of a laboratory. A programme and a setup to implement this project has been developed at this. A results gathered in Chapter 4 Result and Analysis documented throughout this stage will be more reliable, proven and realistic if the content of this document is demonstrated on the real wind blade that located in Pilot Plant area. However, to develop the full functionality of this project, a person also needs to understand the transmitter and receiver aspects of the project. The establishment of the basic data logging for the main project was achieved based on the result collected.
Acknowledgement

I would like to express my appreciation to my supervisors, Dr Gareth Lee and also Dr Jonathan Whale that provided support and shared their knowledge for this project. Not just for this project but throughout my entire journey at Murdoch University. For their advice, patient and time during my difficult time. I also wanted to say thank to Mr Mark Burt and Iafeta ‘Jeff’ Laava for their help and assistance. Without all these people, I could not completed my degree and achieved more.

I also like to thank all my friends that supported me throughout the entire degree. For their knowledge and help, joy and laughter that we all shared together. Not to forget, to all my family members that always gave me support and were always being there whenever I needed them.

Lastly, thank you to everyone who encouraged, motivated me through my difficult times.
### Glossary

<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ROTA</td>
<td>Renewable Energy Outdoor Test Area</td>
</tr>
<tr>
<td>SWT</td>
<td>Small Wind Turbine</td>
</tr>
<tr>
<td>LWT</td>
<td>Large Wind Turbine</td>
</tr>
<tr>
<td>IEC</td>
<td>International Electrotechnical Commission</td>
</tr>
<tr>
<td>ADC</td>
<td>Analog to Digital Converter</td>
</tr>
<tr>
<td>PGA</td>
<td>Programmable Differential Input</td>
</tr>
<tr>
<td>V</td>
<td>Voltage</td>
</tr>
<tr>
<td>$R_G$</td>
<td>Resistor Gain</td>
</tr>
<tr>
<td>NC</td>
<td>No Connection</td>
</tr>
<tr>
<td>PC</td>
<td>Personal Computer</td>
</tr>
<tr>
<td>PCB</td>
<td>Printed Circuit Board</td>
</tr>
<tr>
<td>AREF</td>
<td>Analog Reference</td>
</tr>
</tbody>
</table>
# TABLE OF CONTENTS

## CHAPTER 1 INTRODUCTION

1.1 OVERVIEW ......................................................................................................... 1  
1.2 PROJECT AIM ..................................................................................................... 2  
1.3 PROJECT OBJECTIVE ....................................................................................... 3

## CHAPTER 2 LITERATURE REVIEW

2.1 IEC 61400 ............................................................................................................. 5  
2.2 WIND TURBINE LOADS ................................................................................... 7  
2.3 WIND TURBINE LOADS SOURCE .................................................................. 8  
2.4 WIND TURBINE LOAD DESIGN ................................................................... 10  
2.5. MEASURE THE STRESS OF THE BLADE ................................................... 13  
  2.5.1 STRESS AND BENDING MOMENT ...................................................... 14  
  2.5.2 CALCULATION OF BENDING STRESS ............................................... 16  
2.6 THE CASE STUDIES ........................................................................................ 17  
2.7 STRAIN GAUGE SENSOR ............................................................................... 19

## CHAPTER 3 MATERIALS AND METHODS

3.1 HARDWARE ..................................................................................................... 27  
  3.1.1 HX711 ........................................................................................................ 27  
  3.1.2 INA 125P ................................................................................................... 32  
3.2 SOFTWARE ....................................................................................................... 36  
  3.2.1 ARDUINO IDE ......................................................................................... 37

## CHAPTER 4 RESULT AND ANALYSIS

4.1 HX711 LOAD READING .................................................................................. 39  
4.2 INA 125P LOAD READING ............................................................................. 44  
4.3 COMPONENT COMPARISON ........................................................................ 47

## CHAPTER 5 CONCLUSION

.................................................................. 48

## CHAPTER 6 FUTURE WORKS AND RECOMMENDATIONS

..... 50

## REFERENCES

.......................................................................................................................... 52

## APPENDICES

.......................................................................................................................... 54
LIST OF TABLES

Table 1 Loads Source Classification ................................................................. 9
Table 2 Surface Roughness Classification ...................................................... 12
Table 3 Pin Descriptions .................................................................................. 29
Table 4 Gain Table ............................................................................................ 33
Table 5 Setup 1 HX711 Maximum and Minimum Bridge ............................... 42
Table 6 Setup 2 HX711 Maximum and Minimum Bridge ............................... 43
Table 7 Single Setup of INA 125P ................................................................. 45
Table 8 Multiple Setups of INA125P ............................................................. 46

LIST OF FIGURES

Figure 1 Wind Profile ..................................................................................... 11
Figure 2 Structure of Wind Blade ................................................................. 13
Figure 3 Point Load Bending Moment .......................................................... 14
Figure 4 UDL Bending Moment ................................................................... 15
Figure 5 Varying Load Bending Moment ....................................................... 15
Figure 6 1-Gauge Method ............................................................................ 16
Figure 7 2-Gauge Method ............................................................................ 17
Figure 8 Wheatstone bridge ......................................................................... 19
Figure 9 Balancing the Bridge ..................................................................... 22
Figure 10 INA125P Bridge Setup ............................................................... 23
Figure 11 HX711 Bridge Setup ..................................................................... 23
Figure 12 Base Platform ............................................................................... 26
Figure 13 HX711 ........................................................................................... 27
Figure 14 HX711 Block Diagram ................................................................. 28
Figure 15 Chip on HX711 Board ................................................................. 29
Figure 16 Region Illustration ....................................................................... 30
Figure 17 Horizontal wind Blade ................................................................. 30
Figure 18 HX711 Wiring Configuration ....................................................... 31
Figure 19 INA125P Instrumentation Amplifier ........................................... 32
Figure 20 Scaling ............................................................................................................ 35
Figure 21 Load on Metal Plate ...................................................................................... 36
Figure 22 Arduino IDE Sketch ...................................................................................... 37
Figure 23 Calibration Result ...................................................................................... 39
Figure 24 Calibration Code and Serial Monitor Reading ............................................. 40
Figure 25 140g Load .................................................................................................... 40
Figure 26 470g Load .................................................................................................... 41
Figure 27 Setup 1 Maximum Output, 40mV ................................................................. 42
Figure 28 Setup 1 Bridge at Balance, 0mV ................................................................. 42
Figure 29 Setup 1 Minimum Output, -40mV ............................................................... 42
Figure 30 Setup 2 Maximum Output, 40mV ............................................................... 43
Figure 31 Setup 2 Bridge at Balance, 0mV ................................................................. 43
Figure 32 Setup 2 Minimum Output, -40mV ............................................................... 43
Figure 33 Scaling Illustration ..................................................................................... 44
Figure 34 Single INA125P Max Analog ...................................................................... 45
Figure 35 Single INA125P Balance Bridge ................................................................. 45
Figure 36 Single INA125P Min Analog ....................................................................... 45
Figure 37 Multiple INA 125P Max Analog ................................................................. 46
Figure 38 Multiple INA125P Balance Bridge .............................................................. 46
Figure 39 Multiple INA125P Min Analog ................................................................... 46
CHAPTER 1

INTRODUCTION

1.1 OVERVIEW

This is a continuation project from a previous final year student [1]. There are a few sections that need to be discovered for this project. A wind turbine blade is a blade that attaches to the rotor hub. It is used to rotate the mechanical shaft of the nacelle system and converted mechanical energy into electrical energy using a generator. The rotation of the rotor comes from the harnessing of the wind power.

In this project, the primary interest is the bending moment of the blade for upwind and downwind situations. Upwind is where the wind that flows hits the blades first and then it will pass through other components such as nacelle and tower [2]. Meanwhile, for the downwind case, the wind will hit the nacelle components and tower first before hitting the wind blade [2]. In steady environmental conditions, the blade will usually produce a steady output. But, when the wind speed changes over time, the wind blade will experience an increase in load. Another situation is when the wind blade experience turbulence that is caused by fluctuations in the wind speed. Both conditions create significant interest in the load that the blade experiences. Turbulence is linked with fatigues load damage and thus it is of importance to measure loading on turbines in turbulence sites since it is linked with fatigue damage when turbines are in turbulent sites.

The bending moment of the wind blade can be classified into two categories known as flapwise and edgewise bending. Flapwise bending moment is influenced by the aerodynamic loads resulting from the design of the blade [3]. Meanwhile, for the edgewise bending moment, is influenced by from the blade mass and also gravity.
The bending moment of the wind blade might destroy the wind turbine system, environment and also could cause fatalities [4]. A few factors that influence the bending moment and affect the performance of the wind blade are listed below [3].

1. Blade plan shape and number of blades
2. Aerodynamic design
3. Angle of twist
4. Power regulation

To analyse the bending of the wind turbine blade, usually, a blade load test will be performed. Based on a previous thesis report by Pleiksna, a static flapwise blade root bending moment with load estimation of 300kN has been achieved for 5kW wind turbine blade at the Renewable Outdoor Test Area [1]. They managed to plot the output of the static measurement for the upwind and downwind gauge output [1].

It is difficult to measure the root blade stress on an operating wind turbine in the field without develop the real simulation in a controlled environment first. The system might fails and lead to a wrong data collection. By testing the system on the wind turbine root blade in a laboratory might eliminate the error and faulty that appears. Besides that, it is difficult to investigate the behaviour of the load that applied to the wind blade without having a basic interpretation of data collection.

1.2 PROJECT AIM

Below are the initial aims and objectives for this project that were considered. The main aim for this project was to investigate the impact of turbulence on blade fatigue loading. To achieve the main aim, a few objectives were set to ensure the project moved forward.
1.3 PROJECT OBJECTIVE

The scope of this thesis is to develop a model that represent the actual dimension of the real project by using a flexible beam (representing a wind turbine blade).

The objectives that were set at the early stage of this project are as per below.

1. To implement the previous thesis project to measure the flapwise bending moment.
2. To correlate flapwise bending moment with wind speed and turbulence.
3. To establish data logger communication from the strain gauge sensor installed at the root of the blade to record the measurements of flapwise bending moment.
4. Other than that, the components selection should match the requirement of the project that it needs to be inside of the wind turbine nose cone.

To achieve all these objectives, aims and apply it to the real world, it first needs to be developed in a laboratory. It will simplifies the research problem by developing a strain gauge reading and communication by trying out the system in a controlled environment.
CHAPTER 2

LITERATURE REVIEW

This chapter aims to provide an overview of the loading cases for a small wind turbine including a design that needs to measure the stress based on the load that the blade experiences. It also discusses a case study that is related to this problem. To analyse the bending of the wind turbine blade, usually, a blade load test will be conducted. Based on a previous thesis report by Pleiksna, the root blade load has been performed during the installation of the strain gauge on the 5kW wind turbine blade at the Renewable Outdoor Test Area (ROTA) [1]. The bending of the wind turbine blade also could be analysed under other specific load sources such as aerodynamic, gravitational, centrifugal, gyroscopic and even in normal operational conditions. All of this research is a useful background to the wind blade bending analysis.

An area of interest for this thesis is the gyroscopic force that occurs on the wind turbine blade. A gyroscopic effect is a result of a yawing process of the wind turbine yaw control [5]. The gyroscopic force occur for both horizontal axis wind turbine and also on the vertical axis of the wind turbine. By adjusting a system parameter, it will reduce the effect of the gyroscopic effect. The vibration of wind turbine blade is also one of the sources that lead to it. Installation of the gyroscopic sensor might help to measure and record the effect.

On the other hand, another important thing to consider is fatigue loads. Fatigue load results from the cyclic gravitational load [3]. Analysing the fatigue load of the wind blade design is important to increase the lifespan of the blade significantly.

In this thesis, all the measurements from the sensor installed on the wind blade will be transmitted over to the receiver via telemetry communication. It is vital to
review the previous communication design to check if the objective set will be achieved. The measurements come from strain gauge with 9 degrees of freedom that will be fed to the transmitter installed in the nose section of the wind turbine [1]. The data is then collected by a receiver over a specified range of distance. The previous thesis claimed the experiment was able to reach a 10 m range [1].

For the time being, the system components are not connected to each other. More work needs to be done before it becomes a complete system. The signal from the strain gauge must undergo amplification and a filtering process before being into the transmitter board.

2.1 IEC 61400

Increases in energy demand cause many people try to find the alternatives to obtaining energy. The professional agencies are doing a lot of research to enhance the electrical generation. Green energy is one form that is rapidly growing to provide and meet the demand. Other than being a free source, it also provide much benefit to the society. In addition to this, International Electrotechnical Commission is an international agency, providing a standard requirements and technical descriptions that are followed worldwide. IEC has developed a method and procedure based on the conducted experiment,

IEC 61400 (IEC2013) is an international standard for a wind turbine. From a small to a large wind turbine. In that standard, many guidelines have been introduced to ensure proper installation and safety regulations which have been followed by many people around the world. IEC 61400-1 is mainly for a big wind turbines that can withstand for a higher wind speed while IEC 61400-2 is for a small wind turbines. However, there are many more investigations of large wind turbines as compared to small wind turbines.
Over time, due to the increase in energy demand, the use of wind turbines has expanded into urban areas in which a large and tall buildings sit next to each other. These particular sites create high turbulence intensity. This turbulence creates a significant impact on the wind blade. IEC 61400-2 standard was created based upon an open terrain turbulence environment or standard turbulence model [6]. This standard is not suitable for high turbulence operation. Those non-open terrain environments are usually characterised by highly turbulent wind flow [6]. The highly turbulent site correlates with the turbulence intensity of wind that will reduce the life of the wind turbine.

Many researchers have studied the structural and fatigue loading of large and medium wind turbines based on a different parameters such as the surface roughness, atmospheric instability, and wind speed of a highly turbulent site that may affect the fatigue and life of the wind turbine. However, there is less investigation on small wind turbines that has a swept area of less than 200m² [7].

A report by the National Small Wind Turbine states that only 5 out of the 36 small wind turbines meet the standard. [8].
2.2 WIND TURBINE LOADS

As detailed in “Wind Energy Explained”, wind turbine loads are strictly related to the energy production of the wind turbine [9]. The loads are the critical key to assess the structural integrity of the wind turbine. For the wind turbine, load is defined as the forces or stresses that act towards the turbine. The loads that the wind blade will experience can be:

1). Steady
2). Cyclic
3). Transient
4). Stochastic
5). Resonance-induced loads.

This thesis will focus on the constant load but it is still essential to understand the overall types of load that can exists.

Steady loads will cause a uniform load on the machine. It is a non-time varying load. For the static loads, it is divided into static or rotating. A steady load is sustained by a steady wind flow. The cyclic load is a load that exists due to the rotation of the wind turbine. It varies and is periodic in time. This load is also affected by the weight of the blade, wind shear, and also from the yaw movement. On the other hand, the transient load is a time-varying load that is affected from the outside temporary event such as start, stop and brake application. The shaft will have a different transient loads based upon the number of the blade that rotate in a complete revolution.

Turbulence that results from the variation in wind speed will lead to a stochastic load. The changes of the stochastic loads will affect the aerodynamic forces of the blade. The resonance-induced loads are caused by a natural frequency of a particular component of a wind turbine. It is a cyclic load that results from a dynamic response of
the part. It is crucial in designing the wind turbine to prevent the resonance-induced load from occur.

This thesis will focus on the signal conditioning of static loads obtained from a sensor which is a strain gauge interfaced to the microcontroller. The wind turbine is a 5kW Aerogenesis wind blade. The blade is currently in the pilot plant area. The other aspect to mention is that the experiment will be conducted in a controlled environment area. The controlled environment will help in controlling the disturbance and also provide better access to the project for a quality result. The load that applies to the wind blade is the absolute weight. The resulting data that gathered from the experiments could be used for further investigation.

2.3 WIND TURBINE LOADS SOURCE

The different types of wind turbine loads was just discussed. The various loads come from different sources. The four primary sources of load come from aerodynamic, gravity, dynamic interaction and also from the mechanical control.

Aerodynamic effects are one of the significant sources of load for the wind turbine. The aerodynamic design of the blade leads to a different kind of load such as fatigue damage [3]. The lift and drag coefficient in this, however has a significant role. Below is the calculation involved in the lift and drag force.

Lift force, $F_l$: $\frac{1}{2} C_l \rho v^2 A$

Where; $C_l$= lifting coefficient

$\rho$ = density of air (kg/m$^3$)
$v$ = flow velocity (m/s)
$A$ = body area (m$^2$)

Drag force, $F_d$: $\frac{1}{2} C_d \rho v^2 A$
Where; \( C_d \) = lifting coefficient

\[ \rho = \text{density of air (kg/m}^3) \]

\[ v = \text{flow velocity (m/s)} \]

\[ A = \text{body area (m}^2) \]

Thrust Power: \( P (W) = F_d \cdot v \) \[10\]

For a large wind turbine, the weight of the blade and the tower has a large impact on the installation design of the wind turbine and tower. In summary, the loads sources can be classified as shown in Table 1 below.

<table>
<thead>
<tr>
<th>Loads</th>
<th>Load Category</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mean Wind Rotation</td>
<td>• Steady Loads</td>
</tr>
<tr>
<td>Wind Shear Yaw error and motion Gravity Rotation</td>
<td>• Cyclic Loads</td>
</tr>
<tr>
<td>Turbulence Rotation</td>
<td>• Stochastic Loads</td>
</tr>
<tr>
<td>Gusts Starting/Stopping Pitch motion Teeter</td>
<td>• Transient Loads</td>
</tr>
<tr>
<td>Structure and excitation Turbulence Rotation</td>
<td>• Resonance-induced loads</td>
</tr>
</tbody>
</table>
2.4 WIND TURBINE LOAD DESIGN

The loads that the wind blade experiences are externally supplied either by applied forces or moments of the wind turbine components. The wind turbine components are specifically designed for two types of loads which are ultimate loads and fatigue loads.

In this case, the ultimate loads are considered to be the maximum load the blade can withstand. The fatigue loads refer to the strength of the wind turbine component to withstand a number of cycles of different wind magnitudes.

Take to loads into account, a designer will refer to a particular design. Usually, they will refer to the standard design as mentioned before. This is to ensure that it meets certain set criteria. The priority is given to a few different criteria which includes normal conditions, extreme conditions and fatigue. This is to ensure that the wind turbine can operate safely and can produce larger amounts of energy.

The IEC 61400-1 standard lists three different classifications of wind conditions. These three classifications correspond to different classes of turbulence.

The increases in height will result in different wind speeds and also changes in the turbulence flow. Some designers use software to analyse and simulate the wind turbine. LOADS, YawDyn, MOSTAB, SEACC, GH Bladed and AERODYN are some of the different software package used to do the analysis [3]. The wind profile in Figure 1 below illustrates the effect of the surface roughness on the wind profile.
As can be seen above, the wind profile is different at each level of height. It also varies according to surface roughness, and the wind speed. At the different heights, the wind speed changes due to the obstacles that may exist. In this case, the barriers that exist create turbulence that affects the energy production of the wind turbine.

\[
V_2 = V_1 \frac{\ln\left(\frac{H_2}{Z_0}\right)}{\ln\left(\frac{H_1}{Z_0}\right)}
\]

Notation:  
1). \(V_1\) is a reference wind speed, measured at reference height \(H_1\).  
2). \(V_2\) is the wind speed that located at \(H_2\).  
3). \(Z_0\) is the surface roughness according to the Table 2 below.

Above is the formula used to calculate the wind profile. The method considers the obstacles and surface roughness as a factor that change the pattern. As can be seen in the wind profile illustration Figure 1 above, as the height increases, the wind speed also increases due to the reduction of the obstacles that exist. Hence it will increase the energy production. But, notice that as the height increases, the blade rotor could also experience more turbulence due to fact that the obstacles that may exist in the surrounding environment.
<table>
<thead>
<tr>
<th>Roughness class</th>
<th>Roughness length (m), $Z_0$</th>
<th>Land cover types</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0.0002</td>
<td>Water surfaces like sea and lake</td>
</tr>
<tr>
<td>0.5</td>
<td>0.0024</td>
<td>Open area with smooth surface, concrete, airport</td>
</tr>
<tr>
<td>1</td>
<td>0.03</td>
<td>Open agricultural land without fences</td>
</tr>
<tr>
<td>1.5</td>
<td>0.055</td>
<td>Agricultural lands, building and 8m high hedges separated by more than 1km</td>
</tr>
<tr>
<td>2</td>
<td>0.1</td>
<td>Agricultural lands, building and 8m hedges separated by 500 meters</td>
</tr>
<tr>
<td>2.5</td>
<td>0.2</td>
<td>Agricultural lands, many trees, bushed and plants, 8m hedges and separated by 250 meters</td>
</tr>
<tr>
<td>3</td>
<td>0.4</td>
<td>Towns, villages, agricultural area with many or high hedges, forests and rough, uneven fence.</td>
</tr>
<tr>
<td>3.5</td>
<td>0.6</td>
<td>Large towns with tall buildings</td>
</tr>
<tr>
<td>4</td>
<td>1.6</td>
<td>Large cities with high building and skyscrapers</td>
</tr>
</tbody>
</table>

Table 2 above is the classifications of the surface roughness class that affect the turbulent with the examples provided in the table. The highest surface roughness are coming from a large and high building that usually located in a cities. Hence, this area has the highest turbulent.
2.5. MEASURE THE STRESS OF THE BLADE

Structural load is related to the flapwise and edgewise bending. Structural load is the analysis on the wind blade stress or for simplicity, the integrity of the wind blade structure. Initially, the wind blade analysis will be presented by using the traditional method, with the wind blade modelled as a simple cantilever beam and the stress analysis is done along the section which is a tip, mid and also at the root of the blade. However, with the help from various structural load analysis software packages such as Finite Element Method (FEM), it could be analysed in three-dimensional views providing an accurate result [3].

In this thesis, the stress analysis may done by using a traditional method. The wind blade is positioned horizontally and the stress analysis will be conducted accordingly.

![Figure 2 Structure of Wind Blade](image)

Figure 2 above shows the overall structure of the blade. The structure of the wind blade is divided into three categories, as listed below:

- **The tip** – This is the critical section in aerodynamic. This is a section where the lift and drag ratio will be highly utilised.
- **The midspan** – Provide the maximum allowable aerofoil section that maximises the lift to drag ratio.
• The root - This is the area that connects the hub of the rotor to the first aerofoil profile. It consists of a thick aerofoil section and carries a low aerodynamic efficiency. It experiences the highest load compared to the other part.

2.5.1 STRESS AND BENDING MOMENT

Stress is defined as a particular force that applies to a certain specific cross-sectional area and is measured in Newtons. The applied force causes the deformation of a material [11].

The stress applied to the blade will cause the blade to experience a bending moment. The bending moment is the amount of bending that occurs on a beam. The location and the type of load that exerts on the blade have a significant impact on the amount of bending [12]. There are a few different bending cases such as below.

• Bending moment caused by point load
• Bending moment caused by uniformly distributed load (UDL)
• Bending moment caused by varying load.

Above are just a few types of load cases. But this thesis, will focus on the finding of the bending moment caused by a point load. The calculation of the different load cases as shown below. Consider the blade is in cantilever beam position.

Case 1: Bending moment cause by point load

Figure 3 Point Load Bending Moment
- Bending moment calculation = $W \cdot L$
  \[ = WL \]
The bending moment for this case also depends on the distance of L from a fixed end bending [12].

Case 2: Bending moment cause by uniformly distributed load (UDL)

Case 3: Bending moment caused by varying load.

For this case, the bending moment calculation is:

Bending moment calculation: Area of load diagram*distance of it centroid from the point of the moment [12]
2.5.2 CALCULATION OF BENDING STRESS

Section 2.5.1 above shows the illustration for the calculation for the bending moment calculation. In this thesis, the chosen sensor is a strain gauge. So, to calculate the bending stress, that configuration should be considered as well. There are two calculations involved due to different strain gauge setups. The first one is 1-Gage method and the second one is the 2-Gauge Method.

1). 1-Gauge Method

![Figure 6 1-Gauge Method](image)

The blade will be treated like a cantilever beam. The formula involved to calculate the bending stress for this method is expressed below.

Surface stress,

\[ \sigma = \varepsilon \cdot E \]

\[ M = W \cdot X \]

Where : X is a distance from the position “W” to the centre of the strain gauge

The stress at the moment,

\[ \sigma = M / Z \]

\[ M = Z \cdot \varepsilon \cdot E \]

To calculate Z,

\[ W = b \cdot h^2 \cdot E \cdot \varepsilon / 6 \cdot X \]

b: width of the beam
h: height of the beam
2). 2-Gauge Method

For the 2-Gauge method, the strain is located on the top surface and also at the bottom surface, but both of it must be parallel to each other. By using this method, the output signal to the microcontroller will be double that using just one strain gauge.

To calculate the stress by using this method, the calculation is as below:

\[ \sigma = \varepsilon \cdot E / 2 \]

2.6 THE CASE STUDIES

In 2011, there was an investigation by a master student on a small wind turbine to decide whether the turbulence model from the existing small wind turbine standard is suitable or valid for rooftop sites done in Murdoch University [8]. The investigation was supervised by Dr Jonathan Whale. The case study involved two different places which were Ostergarnsholm in Sweden and also on the rooftop of Bunnings warehouse in Port Kennedy, Perth WA.

The primary objective of the investigation was to focus on the validity of the standard model of a small wind turbine by concentrating and determining the characteristic of the turbulence intensity in open and urban areas at different heights. The findings then compared the normal turbulence model in standard design to that used in the Kaimal spectra method.

From the results gathered, the researcher found that for the open space environment, the turbulence intensity is usually below the normal turbulence model.
from the IEC standard. Meanwhile, in urban areas, the result shows that at the lower speed, the turbulence intensity is lower than the NMD and for the higher speed, the turbulence intensity is higher than NMD. So, it is clear that the turbulence intensity in an urban area is much higher than in open terrain.

In the standard design, the turbulence intensity is considered to be constant but in this case, it behaves opposite due to a complicated situation.

In 2015, researchers extended the investigation of the standard design of IEC 61400-2 using the same location, Port Kennedy, Western Australia. In this investigation, the power spectra of the turbulence intensity from the measurement was compared to the calculation made from the von Karman and Kaimal model. This investigation is used to predict the structure turbulence for a SWT in an urban area.

Through the modification of the Kaimal spectra, the researcher managed to model the structural loading of a SWT in 2016 [6]. In this research they studied the accuracy of the aeroelastic modelling to predict the loading on the wind turbine in an urban area.

Finally, in 2017, they showed that the international standard IEC 61400-2 under predicted the level of turbulence specifically at the Port Kennedy site. From the measured data, it shows that in an urban environment, the turbulence is higher by 18%. A higher turbulence results in high power. But due to the turbulence, the fatigue loading experienced by the wind turbine is much higher. From the research done, the turbine experiences a significant high loading compare to the IEC 61400-2 standard for about 58%. For a small wind turbine, this will shorten the nominal lifespan of the turbine itself [13].
2.7 STRAIN GAUGE SENSOR

![Wheatstone bridge](image)

Figure 8 Wheatstone bridge

Figure 9 above is an illustration of the Wheatstone bridge setup of the strain gauge. The Wheatstone bridge has multiple resistors. The strain gauge resistor value installed on the blade that located in ROTA was 1000Ω. But, during the implementation of this project, the strain gauge that has ordered had a different resistor value of 350Ω and 120 Ω. This is due to the fact that the time taken for the same strain gauge that was used in ROTA to be ordered took a long time to arrive. Even though the strain gauge that installed on wind blade in ROTA has 1000 Ω, this same principle could be used. Check the output voltage of the bridge that has 1000 Ω resistor value to ensure that the output signal could be read by the amplifier.

The sensor that was used in this project involves a combination of multiple strain gauges to form a Wheatstone bridge. A Wheatstone bridge will be installed on the wind blade to detect the deflection and bending that occurs resulting from the loads that are applied to it.

The results of a deformation of material caused by the applied load are called strain. [14]. A strain is defined as the ratio of changes of the different strain gauge material length to the original length is given by formula below. [14].

\[
\text{Strain, } \varepsilon = \frac{\Delta L}{L}
\]

\[
\Delta L = \text{difference in length}
\]

\[
L = \text{the original length of the strain gauge material}
\]
Each strain gauge has a different sensitivity to a strain, which is known as the Gauge Factor. A particular gauge factor value represents specific types of strain gauge. It could be obtained from a manufacturer datasheet or a calculation as detailed. However, for this project, the gauge factor for the strain gauge is 2.11. Most metallic strain gauge factors are around 2.

\[
GF = \frac{\Delta R/R}{\Delta L/L} = \frac{\Delta R/R}{\varepsilon}
\]

Strain is proportional to the changes of the electrical resistance value when the force is applied to the wind blade. The strain measurement usually involves only a very small quantity of strain. So, to know the resistance changes, a formula involving calculating the changes is provided as an example below.

Example: FLA-6-350-11-1LJC (Strain Gauge model)

\[
\frac{\Delta R}{R} = \varepsilon \times k
\]

\[
\frac{\Delta R}{R} = \frac{\Delta L}{L} \times k
\]

\[
\frac{\Delta R}{R} = \frac{0.06}{6} \times 2.11
\]

\[
\Delta R = \frac{0.06}{6} \times 2.11 \times 350\Omega = 7.385\Omega
\]

The length of strain gauge metallic coil is 6mm. k is the gauge factor of that strain gauge model and 1% of total length is 0.06mm. So, the sensitivity of 1% changes of metallic coil length gives a resistance value of 7.385Ω. The sensitivity of the strain gauge is important to predict the output voltage of the bridge.

The Wheatstone bridge is the combination of two electrical circuits that are parallel to each other [14]. Both circuits are treated as a voltage divider. The output voltage of the Wheatstone bridge is measured at the centre of the bridge. This output voltage is the output signal that is measures the bending of the load that applies to the blade. The formula below shows the calculation that involves the output reading [14].
\[ V_O = \left[ \frac{R_3}{R_3+R_4} - \frac{R_2}{R_1+R_2} \right] \cdot V_{EX} \]

Where \( V_{EX} \) is the excitation voltage that supplied to the bridge.

According to National Instruments, the suitable voltage range for the excitation voltage of the strain gauge is around 5V-15V [14]. For this project, there are two components that provide the amplification and filtering process as mentioned previously. The HX711 from Sparkfun has a built-in feature that could supply 5V voltage to the Wheatstone bridge, whereas for the INA125P, the chip provides has different range of voltage that could be used to power up the bridge. However, for clarity, the bridge also uses a 5V supply coming from the INA125P chip.

Balancing the bridge is one of the critical aspects in this section. The bridge needs to be at a balanced state to produce a reference reading for a microcontroller to read the bridge output voltage. In this project, a simulation of this process has been done to read the output voltage with the help of a variable resistor. An example of how to balance the bridge can be understood in Figure 10.
At this point, a half-bridge Wheatstone bridge was configured to get the reading from the strain gauge. It would be a waste to install the strain gauge on a metal plate that was only used for getting the signal reading in the microcontroller and test the system. But for the installation on the real wind blade, it needs to be a full bridge configuration. This is due to the deviation of the reading that is caused by the temperature. A full bridge configuration might eliminate and compensate the deviation reading.

Example on balancing the bridge:

- R1 and R2 is set to a fixed resistor value, 200Ω
- Strain gauge value of 350Ω
- A variable resistor or potentiometer
- Excitation voltage at 10V

If \( \frac{R_3}{R_4} = \frac{R_1}{R_2} \) then \( \Delta V = 0 \)

\[ \frac{R_2}{R_1 + R_2} = \frac{R_4}{R_3 + R_4} \]
R2(R3 + R4) = R4(R1 + R2)

\[
R2 \times R3 + R2 \times R4 = R1 \times R4 + R2 \times R4
\]

\[
R2 \times R3 = R1 \times R4
\]

\[
R4 = \frac{R2 \times R3}{R1}
\]

So, in this case, R1 and R2 is a pre-set value, thus; R4 need to be 350Ω to ensure that the bridge is balanced and could produce a reference voltage without having any load exerted on the blade or metal plate on this case.
Figure 8 and Figure 9 shows the implementation of the Wheatstone bridge with other components. Figure 8 shows multiple setups for the INA125P chip while Figure 9 shows a setup for the HX711. As what could be seen in Figure 9, a metal plate is used to simulate the deflection and bending of the wind blade. The metal plate was then clamped to a table. When the bridge is at balance and no load is applied on the metal plate, the reading of the load in the program should display 0 Newtons. This will be discussed later in the program and software section. The bridge needs to be at a balanced state to produce a reference reading for the microcontroller input.

Installation of the strain gauge is not able to be done at this stage. However, information on the installation has been gathered in this project. Below are a few point that should be considered before installing the strain gauge on the wind blade [15].

- Preparation of surface
  
  The surface of the material should be cleared first from any materials such as dirt, grease or paint. Wipe the surface with a wet tissue that has solvent to ensure the surface is free from stain.

- Preparation of strain gauge

  The strain gauge comes with a plastic packet to protect the strain gauge from being broken. Do not remove the strain gauge sensor from the plastic packet before/prior to the installation. Careful care should be taken to handle the sensor. The strain gauge sensor also comes with lead wires for the installation.

- Installation of the strain gauge

  For the installation, determine the location of the strain gauge installation first. In this case, the location would be at the root of the blade. Thus, it will provide a set of results coming from a different location of the wind blade once a specific
load is applied to the blade. Use a tape to hold the upper section of strain gauge before applying glue and place it on the blade surface.

- Adhesives used

Three types of recommended adhesive can be used which are epoxy resin, phenol-epoxy resin and pressure sensitive (Cyanoacrylate series) [15]. But for this type of strain gauge, it needs to be epoxy resin to glue between strain gauge and the blade.

Seek advice from Mr Mark Burt regarding the installation of the strain gauge on the material.
CHAPTER 3
MATERIALS AND METHODS

This section discusses the approach used to implement the project. It also describes the functionality of the components that were used in this project.

![Base Platform](image)

The above picture in Figure 12 shows the base platform that was constructed in this project. At this stage, the base platform has not yet been finished. It is located in the Pilot Plant area. It is important for those who continue this project to start with the mechanical design of the platform. This is because the mechanical design of the platform is related to the loads that will be exerted on the wind blade. Please consult Mr Mark Burt as he is the one who is able to design the mechanical parts, assemble and attach the wind blade to the platform. The maximum load that this blade could withstand is around 30kg which is equivalent to around 294.19 Newtons ~ 300 Newtons. Thus, the load design must follow this rating.
3.1 HARDWARE

The hardware involved in this project is made up from a combination of multiple components. The strain gauge sensor was bought from Showa [15]. Amplifier and filtering includes two different parts which is HX711 module from Sparkfun and INA125P from Texas Instrument [17]. The reason a different components was used for this development is to identify which component will provide accurate reading, robust and can integrate with the system properly. The microcontroller unit for this project was chosen to be an Arduino.

The wiring diagram for all these components is provided in the appendices.

3.1.1 HX711

![Figure 13 HX711](hx711.jpg)

The HX711 is a small module that is created and manufactured by Sparkfun [16]. Figure 13 above shows the image of the HX711 module. Some features include the capability to amplify and filter the signal coming from the strain gauge. Initially, the HX711 is used with a load cell. A load cell is used to measure weight and the configuration of the load cell is made up from a full Wheatstone bridge. It is commonly used for weight scales and industrial process control application. For this project, it has
been chosen because of the functions that it has. The chip module is capable easily interfacing with the microcontroller which in this case is an Arduino.

Figure 14 above shows the block diagram for the HX711 chip module. As seen in the picture, the chip module is made up of a combination of various components. In Figure 14, it shows a load cell is connected to the input of the multiplexer. But, for this simulation purpose, the bridge output is the one that is connected to the multiplexer input. The module has two multiplexer inputs, which is Channel A and Channel B that corresponds to different programmable differential inputs (PGA). Channel A can be programmed to 128 gains (±20mV) or 64 of gain (±40mV). Meanwhile Channel B has a fixed gain of 32. So, a user could select any input and program it in the programming environment.

The HX711 is a 24 bit analog to digital converter (ADC). Compared to the INA125P chip module, the HX711 has a higher resolution. The INA 125P on the other hand only uses a 10 bit ADC that the Arduino microcontroller has. For this project, a gain of 64 has been used because the full differential voltage output from the bridge is required to be ±40mV. In other word, the bridge is getting a 5V voltage supply from the
HX711, and the output must be adjusted to produce 0mV first when the bridge is at balance. When a load is applied to a wind blade or in this case the metal plate, the variation of the load will produce a small voltage output. That small voltage output could be a positive or negative differential voltage.

All the features seen in Figure 15 come from a small 16 pin chip that located is located on top of the HX711 module. Table 4 below provides description of this chip [16].

<table>
<thead>
<tr>
<th>Pin</th>
<th>Descriptions</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Provide power to the board in 2.5V to 5.5V range to the HX711 board.</td>
</tr>
<tr>
<td>2</td>
<td>Act as an analog output regulator.</td>
</tr>
<tr>
<td>3</td>
<td>Provide analog power supply to the bridge around 2.6V to 5.5V.</td>
</tr>
<tr>
<td>4</td>
<td>Act as an analog input regulator.</td>
</tr>
<tr>
<td>5</td>
<td>Connection to ground.</td>
</tr>
<tr>
<td>6</td>
<td>Analog outputs bypass reference.</td>
</tr>
<tr>
<td>7</td>
<td>Negative input for Channel A.</td>
</tr>
<tr>
<td>8</td>
<td>Positive input for Channel A.</td>
</tr>
<tr>
<td>9</td>
<td>Negative input for Channel B.</td>
</tr>
<tr>
<td>10</td>
<td>Positive input for Channel B.</td>
</tr>
<tr>
<td>11</td>
<td>Digital input of serial clock input and power down control.</td>
</tr>
<tr>
<td>12</td>
<td>Digital output of serial data</td>
</tr>
<tr>
<td>13</td>
<td>Crystal I/O</td>
</tr>
<tr>
<td>14</td>
<td>Crystal I/O or input for external clock.</td>
</tr>
<tr>
<td>15</td>
<td>Rate control for output.</td>
</tr>
<tr>
<td>16</td>
<td>Power digital supply 2.6V to 5.5V range.</td>
</tr>
</tbody>
</table>
Figure 16 illustrates the different regions that are involve for this project implementation. When the bridge is at the balance condition, the output voltage of the bridge would be 0mV. If a maximum load is applied to it, will produce ±40mV. In this case, when a pressure is applied on the metal plate, it will deliver a value between these two regions. If the metal plate moves downward, it will give a value within the negative region, and if the metal plate is pulled upward, the output bridge would be within the positive region. Same goes for the reading in the serial monitor of the Arduino. It will display the load that exerts on the metal plate according to the region, so a user could know the movement of the plate either as it moves downwards or upwards based only on the display.

Imagine a wind blade is placed horizontally when a load is applied on top of it, it will move downward and when the load is applied from the bottom of wind blade, it will move upward. When no load is applied on it, it will be in a steady condition. The
positive region in Figure 16 is applicable when the blade is moved upward meanwhile the negative region is applicable when the blade is moved downward. At a steady state, the bridge supposedly to produces 0mV. When a maximum of 30kg or 300N is exerted on the wind blade, the voltage output of the bridge would be -40mV and the reading in the microcontroller will show the value of load that is apply on the blade to be -300N. The negative and positive sign are only used to differentiate between the upward and downward movement of the blade or metal plate.

![Figure 18 HX711 Wiring Configuration](image)

Figure 18 illustrates the connection of the Wheatstone bridge connected to the module. As mentioned above, for this project, the Wheatstone bridge output is between ±40mV. Hence, the module could read it and bring it to readable value by using a gain that could read in the program. The RED and GRN pin on the HX711 board supply power to the bridge, while the BLK and WHT pin on the board is used to receive the output voltage from the bridge. The right side of the board is the connection to the Arduino microcontroller. VDD and VCC are shorted together and connected to 5V pin supply on the Arduino board. While DAT and CLK are connected to the analog pins of the Arduino. A full schematic drawing is available in the appendices sections of this document.
3.1.2 INA 125P

The INA 125P is a 16 pin instrumentation amplifier [17]. INA12P as shown in Figure 19 has precision reference voltages which are 10V, 5V, 2.5V and 1.24V [17]. It has a low quiescent current and low noise. One of the features that this chip has is that its operating temperature can be between -40°C to 85 °C which is suitable for this project and environmental conditions. This INA125P is suitable to be used to amplify pressure and temperature signals. It also can be powered up by using a battery which is one of the requirements that needs to be considered once the overall system is installed and tested. This chip only requires low power and provides high accuracy of amplification readings. The gain of the 125P instrumentation amplifier can be set between 4 to 10000 based on the \( R_G \) setting. The \( R_G \) value could be set on the resistor between Pin 9 and Pin 8. In this project, \( V_{ref} \) is used to provide 5V supply for the bridge.
voltage excitation. The voltage output from the bridge was fed to Pin 6, Vin+ and Pin 7, Vin-. In this project, the configuration of the bridge is not a full Wheatstone bridge yet. But, the approach of the system could use a full bridge configuration. But, the gain and bridge output voltage needs to be checked once the strain gauge sensor is installed on the real wind blade.

To set the bridge at balance, first ensure the excitation voltage. In this project, it has been decided to be 5V. Then, adjust the R4 as in Figure 9 until the output of the bridge gives 0V reading. When, the reading shows 0V, it means the wind blade or in this project, a metal plate is in a steady stationary condition and no load is applied. The output of the bridge then feeds to Pin 6 and Pin 7. The INA125P will amplify the bridge output voltage according to the desired gain needed for the microcontroller to read the INA12P output. In this case, the resistor that is set between Pin 9 and Pin 10 is 10kΩ which means the gain that is needed for the INA12P to amplify the signal voltage is 10. Table 5 below is the gain that could be chosen and the value of the resistor involved [17].

<table>
<thead>
<tr>
<th>Desired Gain</th>
<th>$R_G$ (Ω)</th>
<th>Nearest 1% $R_G$ value (Ω)</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>NC</td>
<td>NC</td>
</tr>
<tr>
<td>5</td>
<td>60k</td>
<td>60.4k</td>
</tr>
<tr>
<td>10</td>
<td>10k</td>
<td>10k</td>
</tr>
<tr>
<td>20</td>
<td>3750</td>
<td>3740</td>
</tr>
<tr>
<td>50</td>
<td>1304</td>
<td>1300</td>
</tr>
<tr>
<td>100</td>
<td>625</td>
<td>619</td>
</tr>
<tr>
<td>200</td>
<td>306</td>
<td>309</td>
</tr>
<tr>
<td>500</td>
<td>121</td>
<td>121</td>
</tr>
<tr>
<td>1000</td>
<td>60</td>
<td>60.4</td>
</tr>
<tr>
<td>2000</td>
<td>30</td>
<td>30.1</td>
</tr>
<tr>
<td>10000</td>
<td>6</td>
<td>6.04</td>
</tr>
</tbody>
</table>
After setting $R_G$ and the gain of INA125P, it is time to check the output voltage of the chip. Make sure the output voltage is at 2.5V when the bridge is at balance. The Arduino microcontroller reads the analog input from 1V to 5V and gives the reading between 0-1023 because it has a 10 bit ADC. When the microcontroller reads 2.5V, it will display 512 on the serial monitor to show that the bridge is at balance, and the load reading is 0N. So, the middle of the microcontroller input is set to be at balance. If a load is applied to a wind blade or a metal plate in this case, the microcontroller will read the state based on the region as mention in Figure 13. The procedure below describes the steps taken during the implementation of this INA125P instrumentation amplifier [18].

1) For $R_G$, use a variable resistor or a multturn trimpot resistor that has a low tolerance value. In this project, the multturn trimpot used had a tolerance of about ±0.5%.

2) After finishing with the strain gauge installation and all the hardware is assembled, it is time to calibrate the output offset and also set the INA125P gain. Output offset is defined as a small voltage that applied is applied to the output of the amplifier and it is required to bring the offset voltage into a readable value, stable and also linear range. The output of the amplifier produces a small, voltage because one single supply is used to power up the amplifier. The reading of the amplifier voltage output could be unstable and inaccurate at this stage even though no load is applied to the wind blade or metal plate. So, as mentioned earlier, when the bridge is at balance, it will produce 0V and the output of the amplifier need to produce 2.5V. By adjusting the voltage that applies to Aref, the output of the amplifier could be brought up to 2.5V for the microcontroller to read as 512 in the serial monitor. Check with a multimeter to
ensure that the amplifier output is correct. The Aref value could be adjusted by turning the variable resistor that connects to Pin 5. Refer schematic in Figure 19.

3) As mentioned before, the gain of the instrument amplifier has been set at 10 by using a 10kΩ resistor. Now, it is time to check that the output of the amplifier gives a reliable voltage reading. Measure the output voltage by using a multimeter. It is important to read the output reading for scaling purposes.

![Figure 20 Scaling](image)

When the metal plate is moved upward or downward, the amplifier output will give the output range between 0V to 5V corresponding to the microcontroller input voltage. The downward movement will give a reading between 2.5V to 0V. While for the upward movement, it will provide a reading between 2.5V to 5V. This scaling has been done in the program section.

4) Double check the output of the amplifier to ensure that it produces 0V when no load is applied. If the output of the amplifier produces not even close to 2.5V at this stage, try to re-adjust the offset output to bring it back close to 2.5V.

5) Figure 18 display a known load that applies to the metal plate after ensuring the offset voltage is correct. A known load is applied on the metal plate to justify
that the bridge produces an output voltage and the amplifier output voltage also gives a readable input voltage to the microcontroller.

![Figure 21 Load on Metal Plate](image)

3.2 SOFTWARE

This section provides a basic understanding of the software and programming environment that was used in this project. An Arduino Mega is the microcontroller which was used in this project. It meets the requirement for this project in that something small was needed for the operating device. For a real project in ROTA, the microcontroller and other devices need to be inside the nose cone of the wind turbine. Only a strain gauge will be installed on the wind blade. The analog signal from the strain gauge will then be transmitted to a receiver. The receiver will log the data that comes from the transmitter with the help from a Datataker to log the data. However, some improvement could be made for this design and will be discussed in the recommendation section.
3.2.1 ARDUINO IDE

The basic Arduino sketch consists of two functions. Figure 22 shows the sketch of the Arduino IDE. Two functions that are involve in the sketch are **setup** function and **loop** function [19]. It usually exists in the default sketch of Arduino IDE. These two functions are the structure of the IDE. This Arduino IDE is an open source software and can be used with any Arduino board. [20]. This project used the Arduino Mega as its main board. For the installation of the Arduino IDE, help was sought from Mr. Will Stirling as he is the officer that was able to install it on any PC in the engineering areas.

The **setup** function only runs one time for every single time the sketch is uploaded to the microcontroller. Meaning that all the statement that are place between the brackets for the **setup** function will be executed for a single time. Compared to the **loop** function, every statement that is placed in between the **loop** brackets will run continuously or in a loop. The execution of the statement takes place first from the top to the bottom of the structure. It is important to have the **loop** function even though no statement written in the **loop** because it is the basic structure of the Arduino IDE programming. If any of the structure are not included, it will lead to a faulty program. With the **loop** function, the microcontroller will run the programming written in between the loop with the execution time stated in the **setup** structure.
Example of the statement in loop functions.

- If…else…if…else case:

```java
If ( x1)
{
    A
} 
else if (x2)
{
    B
}
else
{
    C
}
```

The above statement consists of three different kinds of sections which is A, B, and C. This case has two conditions to be satisfied. If the condition satisfies x1, the program will execute all statements in bracket A, else if it satisfies condition x2 it will execute all the statement in bracket B. If none of the statements are satisfied, the program will then execute the statement in bracket C. This approach was used to create the program and provide more flow to the code execution. All the program code is made available in the appendices section at the end of this document.
CHAPTER 4

RESULT AND ANALYSIS

The results obtained throughout this project can be divided into two categories. These two categories come from different components which are a HX711 module and the INA125P chip. In each category, multiple setups have been used to ensure that the programming code doesn’t create any conflicts between each other and the microcontroller is able to receive and read the output voltage signals from various point of location points.

4.1 HX711 LOAD READING

To be able to read the load correctly, a calibration must be done. The calibration procedure for the HX711 is different compared to the INA125P. Calibration for the HX711 module involves offset adjustment in the programming code. Meanwhile, for the INA125P, the calibration done in the physical setup mentioned in Chapter 3 Methodology, section INA125P. The code used for HX711 calibration is provided at the end of this document in the appendix.
In Figure 24 above, the calibration_factor for the setup is -350120.00 and the reading display shows 0.002 kg. The accuracy of the HX711 reading is up to three decimal places. So, during the calibration process, a user could vary the calibration_factor by pressing ‘+’ or ‘-’ sign to adjust the calibration_factor value and press Send in the serial monitor. Once the calibration_factor is set and given a consistent reading, a known weight is placed on the plate.

Figure 25 shows the results from a 140g load that applied to the metal plate. A similar approach needs to be conducted once the strain gauge sensor is installed on the wind blade.
Figure 26 shows a 470g load that is applied on the metal plate to double check and ensures that the result is reliable and accurate. This calibration must be conducted for every single HX711 used.

As mentioned in the introduction section of this chapter, each HX711 module and the INA12P chip has multiple setups. Tests conducted for each component involve the maximum and minimum value that the bridge could produce and checking the reading of the microcontroller reads. Table 6 and Table 7 below tabulate the results that were acquired. A calculation to convert kg to N has been done in the code to read the result in N units. Supposing, the maximum the wind blade could experience is at 30kg then it will display 294 N ~ 300 N and when the bridge is balanced, it will show around 0 N. But it is quite challenging to get exact accurate readings due to the deviations and tolerances that occur. Explanations and the recommendations to overcome this issue highlighted in the recommendation section at the end of this thesis.
### Setup 1 HX711

#### Table 5 Setup 1 HX711 Maximum and Minimum Bridge

<table>
<thead>
<tr>
<th>When the bridge produce maximum voltage output, 40mV</th>
<th>When the bridge at balance, produce 0mV</th>
<th>When the bridge produce minimum voltage output, -40mV</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image1" alt="Figure 27 Setup 1 Maximum Output, 40mV" /></td>
<td><img src="image2" alt="Figure 28 Setup 1 Bridge at Balance, 0mV" /></td>
<td><img src="image3" alt="Figure 29 Setup 1 Minimum Output, -40mV" /></td>
</tr>
</tbody>
</table>

- Figure 27 displays the result when the bridge output was regulated to the maximum full differential voltage at 40mV. Checked with a multimeter to ensure that the bridge produces the correct value. Otherwise, vary the variable resistor at the bridge. Both reading can be set to read either the flapwise or edgewise.

- When the bridge is at balance, and produces 0mV, the serial monitor displays -5.2N or 500g. Supposedly it should display 0N. The reading tolerance is about 0.0167%. It is important to keep this tolerance as low as possible to get the accurate reading.

- The serial monitor displays the maximum amount of load applied to the metal plate. It supposed to display 300N, but it drifts by about 33N.
Table 6 Setup 2 HX711 Maximum and Minimum Bridge

**When the bridge produce maximum voltage output, 40mV**

| Reading from both sensors: 0.00 N | 224.20 mV |
| Reading from both sensors: 0.00 N | 224.20 mV |
| Reading from both sensors: 0.00 N | 224.20 mV |
| Reading from both sensors: 0.00 N | 224.20 mV |
| Reading from both sensors: 0.00 N | 224.20 mV |
| Reading from both sensors: 0.00 N | 224.20 mV |
| Reading from both sensors: 0.00 N | 224.20 mV |
| Reading from both sensors: 0.00 N | 224.20 mV |

- The right value in the serial monitor is the second setup for HX711 module reading. It also shows the same behaviour as the first one. But both setups can run simultaneously. Figure 30 is used to differentiate between the first HX711 setup reading to the second HX711 setup reading.

**When the bridge at balance, produce 0mV**

| Reading from both sensors: 0.00 N | -4.60 N |
| Reading from both sensors: 0.00 N | -4.60 N |
| Reading from both sensors: 0.00 N | -4.60 N |
| Reading from both sensors: 0.00 N | -4.60 N |
| Reading from both sensors: 0.00 N | -4.60 N |
| Reading from both sensors: 0.00 N | -4.60 N |
| Reading from both sensors: 0.00 N | -4.60 N |

- One of the other factors that might cause the serial monitor not to read 0N is the calibration_factor value. Do not change the position and the clamp thread turn that clamps the metal to the table. Based on experience, it will affect the microcontroller reading.

**When the bridge produce minimum voltage output, -40mV**

| Reading from both sensors: 0.00 N | -322.33 mV |
| Reading from both sensors: 0.00 N | -322.33 mV |
| Reading from both sensors: 0.00 N | -322.33 mV |
| Reading from both sensors: 0.00 N | -322.33 mV |
| Reading from both sensors: 0.00 N | -322.33 mV |
| Reading from both sensors: 0.00 N | -322.33 mV |
| Reading from both sensors: 0.00 N | -322.33 mV |

- Almost the same result in Figure 29. The serial monitor reading drifts by about 20N.
4.2 INA 125P LOAD READING

The INA 125P also has multiple setups. Both of the setups are independent of each other. It can run simultaneously but on a different basis. Table 8 and Table 9 below show some of the results that were gathered during this project.

As can be seen in these two tables, the microcontroller reading is not constant and steady. It could be solved by constructing a filter circuit and applying it to the output of the amplifier before entering the microcontroller input. This will give a more constant and steady reading in the serial monitor.

When the strain gauge bridge produces 0V, the output of the amplifier will produce 2.5V to the microcontroller input. This is because of the scaling region that was implemented in this project as was explained in Chapter 3. When the microcontroller reads 2.5V voltage output from the INA125P amplifier, it will read as 512 and display 0N in the serial monitor. The maximum value of the amplifier could produce was 5V and minimum that it could produce was 0V. This amplifier voltage output plays a significant role because the Arduino microcontroller could only read a voltage value within 0-5V.

![Figure 33 Scaling Illustration](image)
Setup 1 Single INA125P

Table 7 Single Setup of INA 125P

When the bridge produce maximum voltage output, 0.198V

- When the bridge produces maximum voltage, the amplifier output will produce 5V. Thus, the microcontroller will read the voltage given and write the maximum reading at 1023, 5V and 299.41N.

Figure 34 Single INA125P Max Analog

When the bridge at balance, produce 0mV

- At balance, no load applies to the blade or metal for this project. The strain gauge sensor will produce 0V meanwhile the amplifier output will produce 2.5V at equilibrium. The microcontroller at this point will read 512, and display 0N because no load is applied as in Figure 35.

Figure 35 Single INA125P Balance Bridge

When the bridge produce minimum voltage output, -0.2V

- Figure 36 displays the reading at maximum load. But for a different direction which is when the blade moves downward.

Figure 36 Single INA125P Min Analog
Setup 2 Multiple INA125P

Table 8 Multiple Setups of INA125P

When the bridge produce the maximum voltage output.

- Reading in Figure 37 displays two setups for INA125P. The left reading shows the first INA125P setup reading and the second reading shows the second setup INA125P reading. Both display the reading as expected when the amplifier produces 5V.

Figure 37 Multiple INA 125P Max Analog

When the bridge at balance.

- Bridge at balance, the Arduino microcontroller will display 512 and 0.00N. As can be seen in Figure 38, both readings are displayed according to respective load applied.

Figure 38 Multiple INA125P Balance Bridge

When the bridge produce a minimum voltage output.

- Serial monitor will display around -300N when the microcontroller reads a maximum load is that applied on a blade that moves downward.

Figure 39 Multiple INA125P Min Analog
4.3 Component Comparison

Both of the components can produce a good reading for the stress that applied to the metal plate. But, between both components, the INA125P could provide robust reading when the stress applied. The reading fluctuations and jumping could be eliminated by applying a filter as mention in Section 4.2. While the HX711 gives deviation that continuously occur when the load is applied when system is on to collect the reading. The deviation that occur can lead to inaccurate reading. Hence, it is not suitable to be applied for this system.
CHAPTER 5
CONCLUSION

In a nutshell, this project could be used as a stepping stone for the originally proposed main project. This project content needs to be applied to a real wind blade to establish the signal reading from the strain gauge sensor to the microcontroller as only a concept and development of the project has been done at this stage. The implementation of this project could be used to enhance the understanding on the wind turbulence.

The result shows that sensitivity of the HX711 module is much higher than the INA125P chip module. This is caused by a full the differential voltage that is read and the high conversion resolution that it has. Even though the HX711 has high sensitivity, it also has some disadvantages. The readings in the serial monitor for the HX711 shows some deviation occurs before it moves back to the real value that can be expected even though the load that applies to it is not changing. It does not happen all the time, but it might affect the overall result of the sensor reading. Capacitors that the HX711 module has for the filtering process caused the reading not to jump around, which is suitable for the sensor reading.

Meanwhile, the INA125P reading in the serial monitor shows the accurate reading but it does jump around. This is obviously not good. This could be eliminate by applying a filter to the output signal of the amplifier.

It is good to try out applications with the real load applying on the wind blade and comparing both loads reading accuracy.

Overall, this project has achieved certain criteria and could be considered as partially successful based upon the limitations that occurred during the project timeframe. The development of the programming code could be used with multiple
strain gauge sensors. Finally, the demonstration at the end of this project shows that it can display the amount of load that is exerted on the wind blade or a metal plate.
CHAPTER 6
FUTURE WORKS AND RECOMMENDATIONS

A few things could be improved and implemented in terms of design, signal conditioning and system functionality. A future student could also attempt to make the system more robust, user-friendly and dynamic. Below are a few recommendations that might be useful upon starting this project.

1. Mechanical design

For the mechanical design part, Figure 5 shows the current progress of the platform design. It is important to have this platform to first finish because it will disrupt other processes such as the installation of the strain gauge on the wind blade and system testing. Besides that, a load design also needs to be considered. The point of the load that needs to be applied and location of the strain gauge installation must correspond to each other.

2. Component and hardware installation

At this point, all the components were placed on a breadboard. But the installation is not sturdy and causes trouble. It also causes the sensor readings in the serial monitor to fluctuate. To solve this problem, it is possible and better if all the components are soldered together on a PCB board. It will fix all the components and also will be much easier to understand the project implementation.
3. Data logging

   a. Webserver

   Initially, the previous project had demonstrated the telemetry communication and connection between the transmitters to the receiver. But, in case a better upgrade is needed. Usage of a webserver is needed to log all the data then. all the data filtering processes could be made available through a webserver. By using a webserver, a user could also monitor the data collected online through remote access. MySQL is one of the webserver example that could be used with much application. A simple web data logger such as ThingSpeak could be a great startup.

   b. SD card

   Another thing that could simplify the whole project is by using an SD card. The implementation of an SD card will be less complicated and easier in terms of system data logging. As it does not need a telemetry communication.

4. System functionality

A combination of the previous project and this current project may be the best way to do the of the testing the overall system without installing it first. Once the system is fully functional, then it would be good to install it on the wind blade for investigation and data collection. The sampling rate also should be checked once the system is entirely operational.
REFERENCES


2017].


https://startingelectronics.org/software/arduino/learn-to-program-course/01-program-  

APPENDICES

1. Appendix 1 – Single Code INA125P

   Appendix 1 is the code used with only one single INA125P. This code is used to test the reading by using only one INA125P. This is a basic code developed to use with INA125P.

2. Appendix 2 – Multiple Code INA125P

   Appendix 2 is a program that developed based around the single code for INA125P. This code can be used with multiple setup of INA125P. It can be adjusted if another setup needs to be added on.

3. Appendix 3 – HX711 Single Code

   Appendix 3 is a program that used to get the reading of the stress by using the HX711. This code also include the conversion of the signal into a readable Newton value. The amplification gain of the HX711 can be set and written in this code.

4. Appendix 4 – HX711 Multiple Code

   Appendix 4 provides code that used with a multiple setup of the HX711. To use this code, the calibration factor need to be set first. It doesn’t matter either one or more setup is used (Refer Section 4.1).

5. Appendix 5 – INA125P Drawing illustration

   Show the connection and setup that developed for INA125P.

6. Appendix 6 – HX711 Drawing Illustration

   Setup and wiring connection of the HX711 that gives a readable reading in the microcontroller.
Appendix 1: Single Code INA125P

//declaration of pin
const int analogPin = A0;

//declaration of value which we get
float analogValue = 0;
int count = 0;
float voltage = 0;
float scale;
float force;
float sca;
float neg;
int analog;

void setup() {
  //Print the value on serial monitor to see what we are getting
  Serial.begin(9600);
  //As we connect the AREF to ic as reference so we must declare here what reference we are using
  analogReference(INTERNAL);
}

void loop() {
  //read analog value
  for (count; count < 5; count++) {
    analogValue = analogRead(analogPin);
    //when we put analog reference first values are garbage so i put it zero.
    analogValue = 0;
    Serial.println(analogValue);
    //taking value after 100ms
    delay(100);
  }
}
// now read actual value
analog = analogRead(analogPin);
analogValue = analog;
sca = analogValue / 1023;
voltage = sca * 5;

// convert it to voltage
if (analogValue < 512) {
    scale = analogValue / 512;
    neg = 1 - scale;
    force = neg * (-300);
}
else if (analogValue > 512) {
    scale = analogValue - 512;
    force = scale / 512;
    force = force * 300;
}
else {
    force = 0;
}

// serial print data on serial monitor
Serial.print(analog, DEC);
Serial.print("	");
Serial.print(voltage, 2);
Serial.print("	");
Serial.print(force, 2);
Serial.println("N");

// take value after 200ms
delay(200);
Appendix 2: Multiple Codes INA125P

//declaration of pin
const int analogPin = A0;
const int analogPin1 = A1;

//declaration of variables for A0 pin amp
float analogValue = 0;
int count = 0;
float volt = 0;
float scale;
float f;
float sca;
float neg;
int ana;

//declaration of variables for A1 pin amp
float analogValue1 = 0;
float volt1 = 0;
float scale1;
float f1;
float sca1;
float neg1;
int ana1;

void setup() {

//Print the value on serial monitor to see what we are getting
  Serial.begin(9600);

//As we connect the AREF to ic as reference so we must declare here what reference we are using
  analogReference(EXTERNAL);
}

void loop() {

//read analog value
for (count; count < 5; count++) {
  analogValue = analogRead(analogPin);

//when we put analog reference first values are garbage so i put it zero.
  analogValue = 0;
  Serial.println(analogValue);
}
//taking value after 100ms
    delay(100);
}

//now read actual value
    ana = analogRead(analogPin);
    analogValue=ana;
    sca=analogValue/1023;
    volt=sca*5;
//convert it to voltage
if (analogValue<512){
    scale=analogValue/512;
    neg=1-scale;
    f=neg*(-300);
}
else if (analogValue>512){
    scale=analogValue-512;
    f=scale/512;
    f=f*300;
}
else{
    f=0;
}

//now read actual value
    ana1 = analogRead(analogPin1);
    analogValue1=ana1;
    sca1=analogValue1/1023;
    volt1=sca1*5;
//convert it to voltage
if (analogValue1<512){
    scale1=analogValue1/512;
    neg1=1-scale1;
    f1=neg1*(-300);
}
else if (analogValue1>512){
    scale1=analogValue1-512;
f1=scale1/512;
f1=f1*300;
}
else{
    f1=0;
}

//serial print data on serial monitor
Serial.print(ana, DEC);
Serial.print("\t");
Serial.print(volt,2);
Serial.print("\t");
Serial.print(f,2);
Serial.print("N");
Serial.print("\t");
Serial.print("\t");
Serial.print("\t");
Serial.println("\t");
Serial.print(ana1, DEC);
Serial.print("\t");
Serial.print(volt1,2);
Serial.print("\t");
Serial.print(f1,2);
Serial.println("N");

//take value after 200ms
delay(200);
Appendix 3: HX711 Single Code

//This program is followed by calibration of module
//First of all, include the library because this library doesn’t come with Arduino IDE by default.
#include "HX711.h"

//Declaration of variables
float Readings;
float A;
float temp;
float B;

//Designate which pins are using in our case A1 and A0
HX711 scale(A1,A0);

void setup() {
    Serial.begin(9600);
    Serial.println("HX711 scale measuring");
    //Change the gain to increase the range
    scale.set_gain(64);

    scale.set_scale(-380105); //This value is obtained by using the HX711_Calibration sketch
    scale.tare(); //Assuming there is no weight on the scale at start up, reset the scale to 0

    Serial.println("Readings:");
}

void loop() {
    Serial.print("Reading from both sensors: ");
    A=(scale.get_units(5)); //scale.get_units() returns a float
    //As total is value is ranging from -22 to 22 so
    if (A>0)
    {
        //Convert the value into newton
        temp=A/22;
        B=temp*215 ;
        Serial.print(B);
    }
}
else if (A<0) {
    temp=A/22;
    B=temp*215;
    Serial.print(B);
}
else {
    Serial.print(A);
}

Serial.print(" N"); //You can change this to kg but you'll need to refactor the calibration factor
//Moves to new line
Serial.println();
delay(10);
Appendix 4: HX711 Multiple Code

//This code is same as above just additional lines for 2nd module
#include "HX711.h"
//Declares the variables
float Readings;
//Variables for 1st Hx711
float A;
float temp;
float B;
// Variables for 2nd HX711
float C;
float D;
float temp1;
//Pins to assign for 1st module
HX711 scale(A1,A0);
//Pins to assign 2nd module
HX711 scale1(A3,A2);

void setup() {
  Serial.begin(9600);
  Serial.println("HX711 scale measuring");
  Serial.println("Measuring scale using 2 HX711");
  //set gain for both modules
  scale.set_gain(64);
  scale1.set_gain(64);
  //Set calibrating factor for both
  scale.set_scale(-380105); //This value is obtained by using the SparkFun_HX711_Calibration sketch
  scale1.set_scale(-380105); //This value is obtained by using the SparkFun_HX711_Calibration sketch
  scale.tare(); //Assuming there is no weight on the scale at start up, reset the scale to 0
  scale1.tare(); //Assuming there is no weight on the scale at start up, reset the scale to 0
  Serial.println("Readings:");
  Serial.print("Sensor1");
}
Serial.print("/t");
Serial.print("/t");
Serial.print("Senser 2");
}

void loop() {
  //Get the values for both sensors
  Serial.print("Reading from both sensors: ");
  A=(scale.get_units(5)); //scale.get_units() returns a float
  C=(scale1.get_units(5)); //scale.get_units() returns a float
  //Converting to newton for first module
  if (A>0)
  {
    temp=A/22;
    B=temp*215 ;
    Serial.print(B);
  }
  else if (A<0)
  {
    temp=A/22;
    B=temp*215;
    Serial.print(B);
  }
  else {
    Serial.print(A);
  }
  Serial.print(" N");
  Serial.print("\t");
  Serial.print("\t");
  //convert to newton for 2nd HX711
  if (C>0)
  {
    temp1=C/22;
    D=temp1*215 ;
    Serial.print(D);
else if (C<0)
{
    temp1=C/22;
    D=temp1*215;
    Serial.print(D);
}
else {
    Serial.print(C);
}
Serial.print(" N"); //You can change this to kg but you'll need to refactor the calibration_factor
Serial.println();
delay(10);
Appendix 6: HX711 Drawing Illustration