Master's thesis

NTNU
Norwegian University of Science and Technology
Faculty of Information Technology and Electrical Engineering
Department of Engineering Cybernetics

Henrik Skarre Abrahamsen

AUTONOMOUS VTOL DRONE:

TECHNOLOGIES AND ANALYSES FOR CONDITION MONITORING OF EXTERNAL ROTOR BLDC MOTORS

Master's thesis in Cybernetics and Robotics
Supervisor: Mary Ann Lundteigen
June 2023

NTNU
Norwegian University of
Science and Technology

Henrik Skarre Abrahamsen

AUTONOMOUS VTOL DRONE:

TECHNOLOGIES AND ANALYSES FOR CONDITION MONITORING OF EXTERNAL ROTOR BLDC MOTORS

Master's thesis in Cybernetics and Robotics
Supervisor: Mary Ann Lundteigen
June 2023

Norwegian University of Science and Technology
Faculty of Information Technology and Electrical Engineering
Department of Engineering Cybernetics

NTNU
Norwegian University of
Science and Technology

Preface

This MSc thesis is written in collaboration with my current employer Aviant at the Institute of Cybernetics at the Norwegian University of Science and Technology (NTNU), under the supervision of Professor Mary Ann Lundteigen. This thesis represents the final work requirement for my degree of Master of Science in Cybernetics and Robotics.

Working as a drone engineer for the past two years has given me insight into the general challenges of the drone industry. It felt only natural to write my thesis on a problem within this industry. As a mechanically minded cybernetics student, most things about the drone industry excite me. While working many late nights maintaining the drone fleet at Aviants technical facility in Trondheim, I started thinking about how the logistics of maintenance is both costly and has little automation. Calendar-based maintenance seemed quite old-fashioned in contrast to doing some of the most extreme long-distance autonomous drone deliveries in the world. After some research, I stumbled across a whole field of study within maintenance and reliability with the sole purpose of monitoring the general health of components using sensors and data.

I invite the reader to join me in this journey of investigating how to modernize the maintenance approach in small autonomous VTOL vehicles.
Henrik Skarre Abrahamsen

Trondheim, June 5, 2023

Acknowledgements

I would like to express my sincere appreciation to several individuals who have contributed to the completion of this thesis.

First and foremost, I extend my heartfelt gratitude to Professor Mary Ann Lundteigen for her supervision, guidance, and support throughout the research process. I would also like to thank Assistant Professor Viggo Gabriel Borg Pedersen for providing a comprehensive tour of the RAMS lab at NTNU, which enriched my understanding of the field. Additionally, I am grateful to Krzysztof Cisek for lending me a digital oscilloscope, essential for data acquisition and analysis. Finally, I would like to thank Aviant for the opportunity to write my MSc thesis in collaboration with them.

Henrik Skarre Abrahamsen

Trondheim, June 5, 2023

Executive summary

Purpose: This thesis investigates the condition monitoring of external rotor brushless DC motors used in autonomous VTOL drones. The objective was to build and test a prototype for condition monitoring on the Notus platform currently in use by Aviant. The main focus was on selecting and testing sensors suitable for monitoring health indicators of mission-critical components, with a particular emphasis on the external rotor brushless DC motors used to control the drone while in Multi Rotor flight.

Methods and components: Several different types of sensors were tested, including accelerometers, contactless temperature sensors, and Hall effect sensors. A combination of rotational frequency, temperature, and current measurements was tested for monitoring key health indicators such as bearing wear, and propeller faults.

Results: This thesis found that external rotor brushless DC motors for drones offer several challenges in monitoring health through motor current spectral analysis. Knowing the speed, temperature, and motor current signature, both time and frequency domain features show changes for different motor and propeller wear. The temperature has a large influence on the time domain features, but other parameters like an off-center weight on the propeller can be tracked through these features if the temperature of the motor is held steady.

Conclusion: This thesis provides information for drone manufacturers looking to improve reliability and performance while reducing maintenance costs and downtime through effective condition monitoring techniques. By implementing these techniques, manufacturers can detect early warning signs of impending failure and take corrective action before catastrophic failure occurs. Overall, this thesis concludes that effective condition monitoring is the future of drone maintenance, but more research is needed before the method is ready for industry use.

Table of Contents

List of Tables

List of Figures

Abbreviations

Abbreviation	Description
BLDC	Brushless Direct Current
CBM	Condition Based Maintenance
CM	Condition Monitoring
DC	Direct Current
EMF	Electromotive Force
ER-BLDC	External Rotor Brushless Direct Current
ESC	Electronic Speed Controller
FDM	Fused Deposition Modeling
IC	Integrated Circuit
I^2C	I Squared C
MCSA	Motor Current Signature Analysis
MEMS	Micro Electro Mechanical System
PSD	Power Spectral Density
PSU	Power Supply
PWM	Pulse Width Modulation
RAMS	Reliability Availability Maintenance and Safety
RMS	Root Mean Square
RPM	Revolutions Per Minute
RUL	Remaining Useful Lifetime
SDA	Serial Data
SCL	Serial Clock
UAV	Unmanned Areal Vehicle
VTOL	Vertical Takeoff and Landing
WUL	Wasted Useful lifetime

1

Introduction

1.1 Motivation

Aviant is a start-up that develops a multi-purpose platform for drone-based cargo delivery. Some of the transport services require drones to fly over densely populated areas. Strict safety requirements are put in place to prevent the drone from interacting unsafely with the environment. Although the drone's design and control systems include safety features, it is crucial to prevent failures of mechanical components that could lead to critical loss of maneuverability. The main systems currently in place to prevent such failures of mission-critical components are periodic inspection and maintenance intervals. Intervals for inspection, maintenance, and replacement of components are decided based on the number of flights or flight hours. Inspection intervals are chosen conservatively, which often results in more inspections and replacements than necessary, leading to higher operational costs. The advantage of implementing condition monitoring (CM) is the ability to track degradation or drifts in performance without having personnel inspect the components. By analyzing sensor data one can find signatures that indicate the health state of the component in question, or use it to predict remaining useful life (RUL). RUL is a helpful indicator to decide the optimal time for the next inspection or maintenance. The main motivation for finding the optimal replacement,

inspection, or maintenance intervals is usually safety, cost, or a combination of the two [9]. Replacing a part before it is needed increases normal operating costs, failing to replace a part before it fails can be both dangerous and a large unforeseen cost.

Other industries, for example, the car industry, the commercial aerospace industry, and the process industry [17], have been at the forefront when it comes to condition-based maintenance (CBM) strategies. Through the experiences in these industries, it is clear that a CBM strategy combines the cost-effectiveness and reliability that is needed for drone transport. Little research is done on the lifetime prediction of hobby-grade drone components however, a lot of the research from other industries could be used to gain an understanding of component deterioration in drones. By building a test rig the components can be tested on the ground without risking damaging an expensive drone. Doing measurements in a controlled environment without interference from wind also makes the tests more similar from one test to another. Testing in this way leads to more traceability and control when gathering data, and thus it will be easier to see the effect of the changes made to the components.

1.2 Objective and tasks

This thesis aims to build and test a prototype for condition monitoring on the Notus platform currently in use by Aviant. This will be done by investigating, selecting, and testing sensors suitable for monitoring health indicators of mission-critical components. The main focus will be on the external rotor brushless (ER-BLDC) motors used to control the drone while in multi rotor flight. The data will mainly be gathered from a test rig inside. The ultimate goal is to create a prototype of a CM system and use it to investigate sensors and methods that could be further developed into an onboard solution.

1.3 Delimitations

This thesis will not implement a functioning CM scheme on a flying drone, because of the time limit and the risk involved this is not done. The thesis will leave a

fair amount of time-domain features and frequency analysis techniques unused, a subset of features will be investigated to gain insight into what's possible. There will not be true run-to-failure tests because of time constraints, but rather A-B testing and testing over a slowly varying parameter. Tests will be conducted inside in suboptimal conditions when it comes to moving air with a propeller, and the room will not be temperature controlled. Theory and presentation of rules and regulations will be omitted from the thesis as this is subject to rapid change and is also vastly different from one country to another.

1.4 Research approach

Coming from the field of cybernetics, it was important to get a feel for general maintenance theory. The main source of this knowledge comes from reading [22], papers, articles, and other publications in this field. The findings in some of these publications were first experimented with in the project thesis running up to this master thesis. The merit of the project thesis was to do condition monitoring on a servo motor utilizing suitable sensors. Much of the experience from the previous thesis was utilized in this master thesis, although most things had to be changed because of the new goal. During the project thesis, a visit to the RAMS lab at NTNU guided by Viggo Pedersen was done on the 5th of December 2022. The conversations during this visit gave a good summary of how to do CM in industrial applications, where sensors cost thousands of dollars and the analysis software runs in the tens of thousands per license.

Little research has been done in the field of condition monitoring on small drones in the 1-15kg range. Some papers were found through google scholar and some inspiration was taken from these. Although much of the research is done on pure multi-copters and not Vertical Takeoff and Landing (VTOL) drones, the findings have transferable value as a VTOL is often just a multi-copter with wings and extra motors. This relevance could be more limited when for example a pusher motor is considered instead of a top motor.

Working with this specific drone platform for over a year has given me the insight needed to combine research from different fields and apply it to Notus.

While working on the platform I have developed new mechanical solutions, added sensors, coded new functionality, maintained components, and built new drones from scratch. This has given me a unique in-depth understanding of Notus as a development platform and also great intuition of how different systems on the drone operate, fail, and perform under varying conditions. This knowledge will be used extensively throughout the testing and writing phases of this thesis.

1.5 Previous work

As part of the preparation for this MSc thesis, a project thesis was written during the autumn of 2022. Some of the sections of this thesis are loosely based on and thoroughly improved work already done in the project thesis.

1.6 Structure of the report

After the brief introduction and motivation chapter above, this report will first outline theory for further use in the thesis. Starting with classical maintenance theory and continuing on to present more modern sensor-based solutions. In chapter 3 the relevant theory for the sensors, components, and signal processing will be presented with a special focus on the external rotor brushless DC motor. Chapter 4 will detail the parts of the drone platform in use, the Notus, that are most relevant to this thesis. The chapter will also present some examples of the current state of the failure modes and the maintenance strategies of the Notus. Next in chapter 5 most of the practical work and used methods will be detailed this is also where the system composition of the test rig and its workings will be presented. Further in chapter 6, the results from testing will be presented in chronological order and with adjustments to the testing procedures as new findings are discovered. Lastly in chapters 7 and 8 there will be discussions about the thesis and the decisions made throughout it, before concluding, giving closing remarks and presenting suggestions for further work. The keen reader is also invited to have a look at the appendix for some extra illustrations, and theory that got left out of the main report.

2

Maintenance

This chapter will cover relevant theory of maintenance, faults, failures, and errors both in general and more concretely in the case of drones. Statistical values like mean time to failure and so forth are not provided by the manufacturer of the motor used, thus much theory in the field of statistical reliability theory will be omitted.

2.1 Faults and failures

A failure is an event when the correct or expected behavior of a system is exceeding the acceptable limits. A fault is the inability of a system to perform its intended function [22]. Adding to this, a system can have dangerous failures and safe failures. Dangerous failure makes the system not fulfill its safety-required functions, while a safe failure does not prevent the system from fulfilling the safety-required functions [22]. Faults and failures are for the most part not binary, many of the systems and subsystems in use fail gradually as illustrated in 2.1. Since failures and faults are defined by acceptable deviations and the inability to perform its intended function the same component in different use cases can have different definitions of these states.

There are several different approaches to fault and failure detection, includ-

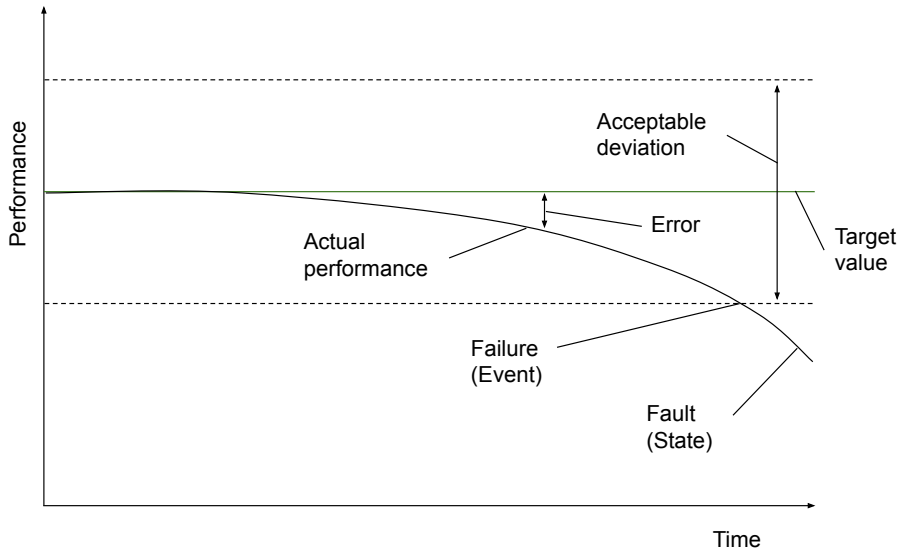


Figure 2.1: Illustration of the difference between failure, fault, and error. Redrawn from figure 3.9 in [22].

ing artificial neural networks and fuzzy logic approaches [27]. These methods all aim to model the behavior of a system in normal operation and compare this to the current state of the system in a way that identifies deviations from normal behavior. Overall, the goal of fault and failure detection is to improve the reliability and safety of systems. By identifying deviations in performance one can predict failures and repair or replace components before they reach a fault state, this is commonly known as preventive maintenance, discussed in 2.2.1.

2.2 Maintenance

According to [22] maintenance is defined as the combinations of all technical and corresponding administrative actions, including supervision actions, intended to retain an entity in, or restore it to, a state in which it can perform its required function. Thus monitoring and sensing key performance characteristics is also part of maintenance. Thus fault detection and analysis of components characteristics can be part of maintenance.

2.2.1 Preventive maintenance

Preventive maintenance is maintenance performed when an item is functioning properly to prevent future failures [22]. The goal of preventative maintenance is to reduce the likelihood of failure in a particular item. The reasons for this can be many, but often it's the desire to reach target reliability. Inspection, lubrication, adjustments, replacement, repair, and calibration can all be part of preventive maintenance [22].

There are many types of preventive maintenance, usually differentiated by the mechanism that governs when the maintenance should occur. The categories include age-based maintenance, based on the age of the component. Age might be measured by time, or other concepts like number of kilometers, number of flights, or flight hours. Clock-Based maintenance, based on specified calendar times. Condition-based maintenance is based on measurements of condition variables that can either be measured continuously or at inspections. And opportunity maintenance, where for example another event blocks the system's function in a way that maintenance will not further increase downtime. All these definitions are presented in [22].

2.2.2 Condition-based maintenance

As a sub-type of preventive maintenance, CBM seeks to measure one or more condition variables of the item at set time intervals or constantly. The measurement of these variables is commonly referred to as condition monitoring. Maintenance is then performed when one or more of these condition variables reach or pass a set threshold. Examples of condition variables include temperature, vibration, and particles suspended in lubrication liquid [22]. Extending this to electromechanical systems we can also include changes in resistance, voltage, or current draw. Condition variables are also the basis of the prediction of the RUL of a component. Incorporating CBM in a system will help prevent faults by replacing parts when abnormalities are detected. Also, it will give components usually subject to clock-based maintenance or replacement the chance to stay in use as long as the component is functioning as intended. If applied in industries where conservative replacement intervals are used as a safety measure, this will drastically reduce the

Wasted Useful Lifetime (WUL) of components while. WUL will also drastically be decreased in cases where the components in use have a wide spread of quality or load in the specific use case in question. As an example using a wheel bearing with a set maintenance interval will need less maintenance if used in a sterile environment than if it is used in an environment where the internals could be contaminated by foreign objects.

A CBM policy requires both the ability to measure condition variables and a mathematical model that can predict the items deterioration process [22]. Furthermore, these condition variables and the mathematical model can also assist in deciding what kind of maintenance is to be done to the item. Doing this analysis in a preemptive manner can improve the logistics of the maintenance. Ordering parts, training personnel, and other preparation can be done before the item is taken out of service reducing downtime drastically.

For condition-based replacements, if $Y(t)$ is a random variable describing the deterioration of an item at time t and it is measured on a continuous scale. The item is inspected in such a way that $Y(t)$ is non-decreasing as t increases. Inspection is done at set intervals, and only then $Y(t)$ is measured. y_p is the lower bound of preventive replacement, y_c is the lower bound of corrective replacement. If $Y(t) < y_p$ at any inspection time $t_1, t_2...$ then the item is left in place. If $y_p < Y(t) < y_c$ then the item is replaced as a preventive measure and lastly if $Y(t) > y_c$ the item is replaced correctively, meaning the item has failed in use. The corrective replacement cost will be significantly higher than the preventive as the item might have stopped production or damaged other items when failing. Replacements are assumed to be perfect such that a replacement brings the item to a *good as new* condition. Example from [22], illustrated in figure 2.2.

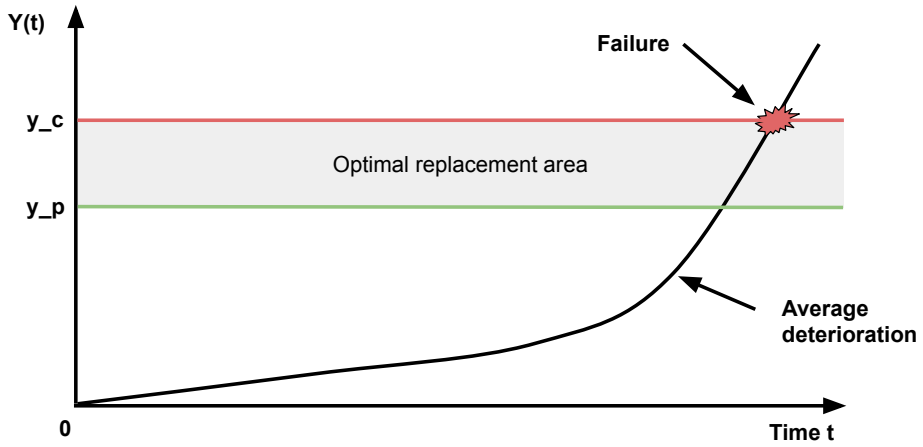


Figure 2.2: CBM illustrated by average deterioration and marks for preventive and corrective replacement respectively, reproduced from [22].

2.2.3 Condition monitoring in drones

Compared to CM in automotive, commercial aviation, and other large technology-driven industries, CM in drones is a relatively new field [16]. In the automotive and aviation industries, electrohydraulic systems are widely used, however, these systems are too large and too heavy to be implemented on drones in the sub-20kg segment. The change from internal combustion engines and jet engines to small fast spinning electrical motors also makes the case different. As discussed in [16], faults in any actuator of an unmanned aerial vehicle (UAV) have the potential to be fatal. This is partly because the UAV is most likely flying when the fault occurs, and partly because there is a tight weight budget for redundancy features. This makes every actuator of the UAV critical to its safe operation in all but the most redundant designs. An obvious difference between quadcopters and a hybrid Vertical Takeoff Or Landing (VTOL) vehicle, is the hybrid VTOL's ability to quad chute if the fixed-wing capabilities are compromised. This redundancy feature does however only work from the fixed-wing state to the multi-copter state. Quad chute is explained in section 4.4.

Conventional fault detection for motors uses electrical or mechanical load. Mechanical monitoring methods can for a UAV create critical problems for the function of the motors in air [16]. Therefore a better approach is using electrical sensors to monitor key features like vibration patterns or current monitoring methods such as Motor Current Signature Analysis (MCSA) [16].

3

Components, sensors and signal processing

This chapter will first off introduce and cover much of the theory regarding the components used in the thesis. Further, there will be an overview of the sensors and their capabilities. Lastly, the theory for signal processing relevant to the goal of the thesis will be covered. The signal processing could cover multiple other statistical time- and frequency-domain analyses, but for this implementation, only the covered subset is used.

3.1 External Rotor Brushless Direct Current Motor

Electrical drive systems demanding high reliability can be found in both industrial and safety-critical applications, such as automotive, aviation, and space industries, either due to potential extreme financial losses or life-threatening safety issues [14]. BLDCs are usually used in critical applications due to their high power density and efficiency in a wide speed range [14]. Although reliable an electrical drive system will not be without failures forever, more on this in section 3.1.2.

ER-BLDC motors are a subset of BLDC motors where instead of the normal

configuration of the rotor inside the stator, the rotor is rotating on the outside of the stator. Some of the big advantages of an ER-BLDC over an in-runner BLDC are the increased torque and more silent operation [21]. To gain comparable torque from an in-runner one would have to attach a gearbox or increase the motor size drastically, both are undesirable for lightweight vehicles with high-reliability requirements.

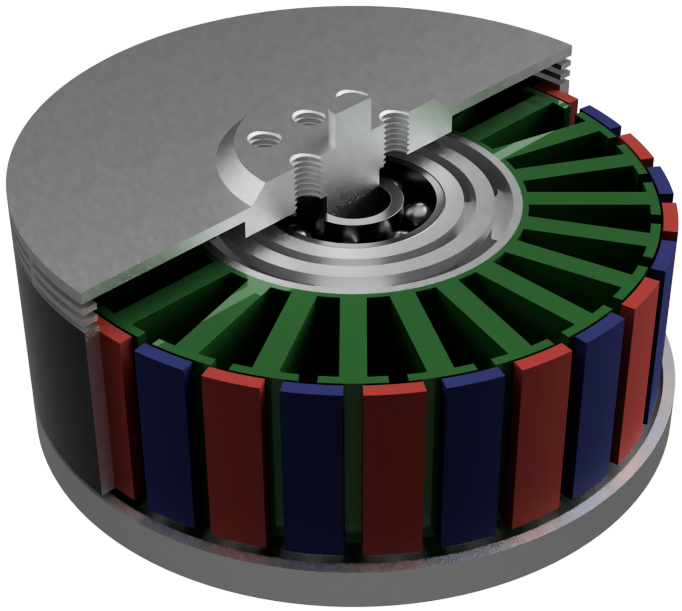


Figure 3.1: Render of T-Motor MN505-S 3d modeled by measurements for visualization, it is not a perfect one-to-one. The rotor is halved to see the slots and magnets of the motor. Windings are not modeled.

Most ER-BLDCs utilize an open-loop commutation system known as an ESC, these work by relating the timing of the next phase activation to the last back Electromotive Force (EMF). This design is referred to as a sensorless design, even though the ESC does in fact sense the back EMF, more on this in appendix B. A closed-loop commutation system uses sensors, often hall effect sensors, on the motor to determine its rotational position. The system then controls the phase activation from the actual position rather than inferring it from another measurement.

Compared to closed-loop commutation the open-loop is inherently less efficient as the back EMF is leading the optimal commutation points by 30 electrical degrees [5]. This leads to a less efficient running motor, more heat buildup, and less constant torque through the rotation leading to excess vibrations, bad or no operation at very low revolutions per minute (RPM), and bad or no de-synchronization recovery. De-synchronization happens when the motor phases are activated at a rotational speed that the physical system can not follow, this will make the motor stop altogether or twitch slightly, either way losing its ability to generate usable torque.

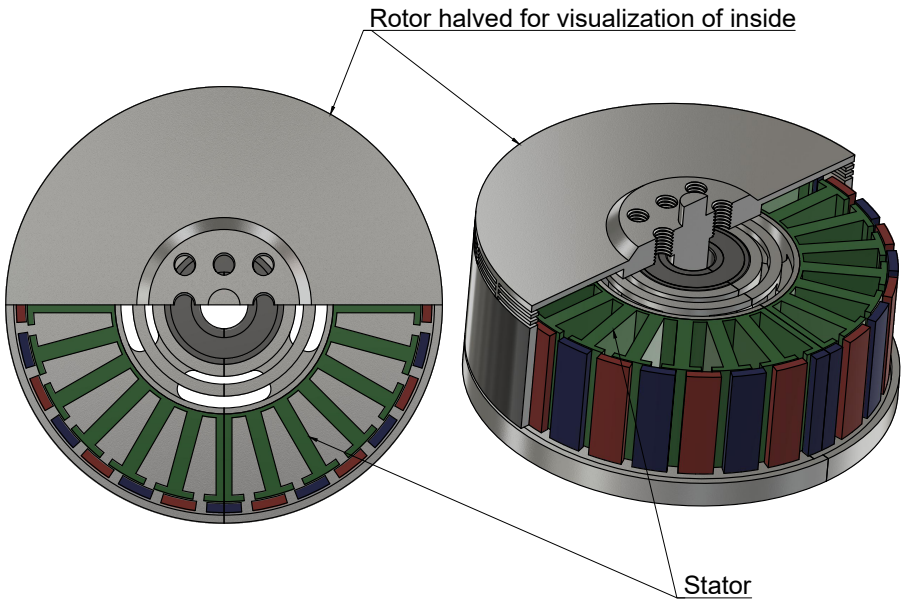


Figure 3.2: Machine drawing of T-Motor MN505-S 3d modeled by measurements for visualization, it is not a perfect one-to-one. The rotor is halved to see the slots and magnets of the motor. Windings are not modeled.

3.1.1 ER-BLDC motors for drone applications

The use of ER-BLDC motors in drones is a testament to their proportionally high torque and efficiency at a wide range of RPM. The ER-BLDC motors are usually larger in diameter, ranging from 2 11cm in diameter, while only being 1 2cm in

axial length [12]. Modern drone requirements do however demand more flexibility, and thus modern drone motors vary a lot in diameter and axial length. The large diameter stems from the substantial torque required in this application, as motor torque is proportional to the square of the radius [12]. These large diameter short axial length motors are often referred to as pancake-style motors by drone enthusiasts. Gaining the torque needed from an increase in diameter is more efficient than gaining the same torque from a transmission system, as these are often heavy and add unwanted noise [12]. Transmissions are also expensive to manufacture and add unwanted complexity. This increased diameter and torque does affect the design of the propellers used on the motors but in a good way. The added torque makes it possible to drive larger propellers, adding to the total efficiency of the system, propellers will be further discussed in section 3.1.3.

3.1.2 Faults ER-BLDC motors

Depending on the application and the design of the motor, different failures can occur. From [4], we find that big industrial induction motors have some typical failure types with a probability shown in table 3.1.2.

Table 3.1: Table collected from [4].

Type of failures	Percentage
Bearing related faults	40%
Stator winding faults	38%
Rotor related faults	10%
Other faults	10%

From experience with the T-motor MN-505S 260KV (3.3) the failures in this ER-BLDC motor also follow a similar pattern. Since T-motor are not sharing their test procedures, test results, or failure analysis, it is hard to know the exact numbers of failure modes. For simplicity, this thesis will focus on faults related to bearing, and rotor/propeller failures as these failures far outweigh the others. Other failures that have occurred while using T-motor ER-BLDCs at Aviant are among others:

- Dust or items stuck in the air gap between the rotor and stator.

- Insulation damage on phase power cables where they attach, leading to a short.
- Aluminium threads yield after multiple propeller reattachments.

These flaws are somewhat dependent on the application and the systems and routines in place when using the motor. There for they will be left out of the thesis. A failure not necessarily dependent on the motor that will be discussed is the failure of propellers, as propeller failure could be measured with the same sensors.

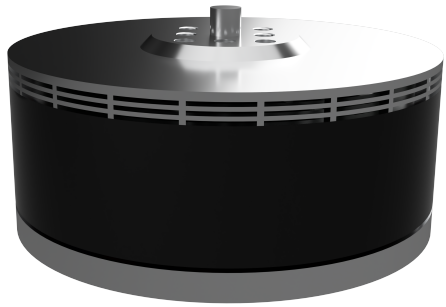


Figure 3.3: T-Motor Navigator MN505-s.

3.1.3 Drone propellers and total system efficiency

Drone propellers come in many shapes, sizes, and blade counts, just like propellers for airplanes. They can be optimized for speed, efficiency, sound, thrust, or for spatial constraints to name a few. This means the usage and the constraints around the system dictate what propeller will be optimal. While there are other constraints, when designing a drone propulsion system the propeller efficiency plays a large role in how efficient the craft is in its entirety. In general propellers with a lower blade count has higher efficiency, but also lower thrust for the same diameter as a propeller with a higher blade count. This loss in efficiency is largely caused by the blades interacting with air that is still turbulent from the last blade passing, more blades, less time between passes, and more interaction. For the same diameter blade [26] found a decrease in efficiency of $\approx 4\%$ for the same thrust in their tests

using a DJI 2212/920KV. Whether the small loss in efficiency is mainly due to the propeller or the motor being optimized for two-bladed propellers is not addressed in [26].

Making wider motors have an impact on the size and shape of the propellers that are used. For example, a wider motor leads to the center of the propeller no longer generating lift, this is however not a large problem as the center of the propeller is the part that generates the least amount of thrust, see figure 14a in [25].

3.1.4 PWM signalling

Pulse Width Modulation (PWM) is a common signaling scheme in use both for servo motors and electronic speed controllers (ESC) in drones. The signaling is done using a change of duty cycle in the signal [23]. A PWM signal is illustrated in figure 3.4.

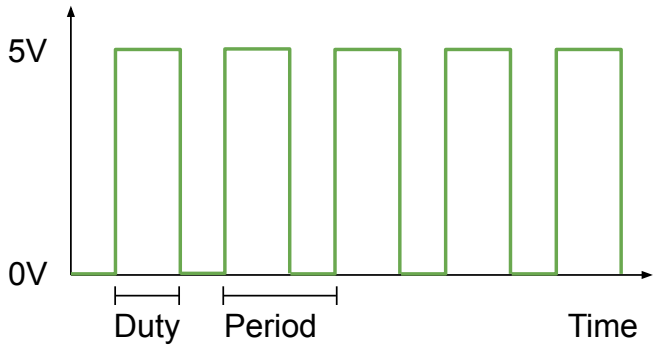


Figure 3.4: Illustration of PWM signal.

3.2 Sensors

3.2.1 I^2C

I^2C is a two-wire data interface with open-drain connections. It uses either 3.3V or 5V logic. I^2C is a master/slave protocol with serial clock (SCL) and serial data (SDA) lines. The I^2C protocol uses a parallel bus and normally has a max data rate of 400kbits/s [23]. This protocol is widespread in the hobby scene where it is used

for almost everything. I^2C was originally designed for internal communication on a circuit board, so problems with transmission speed and stability are normal when the data cables increase in length. Also, the signal is not differential, and therefore it is prone to be affected by interference.

3.2.2 Accelerometer ADXL345

Accelerometers measure acceleration and can use multiple measuring techniques. Most accelerometers are microelectromechanical systems (MEMS) this also holds for the ADXL345 integrated circuit (IC). The working principle is to measure the displacement of a mass attached to a spring as the IC is moved around. One common way to do this is through capacitance change as a mass moves between two conductive plates. To capture acceleration in multiple directions the IC is fitted with one arrangement of mass, spring, and conductive plates for each dimension [23]. This is also the case for ADXL345, the ADXL345 also houses the necessary analog-to-digital converters and digital electronics to send the data as an I^2C signal.

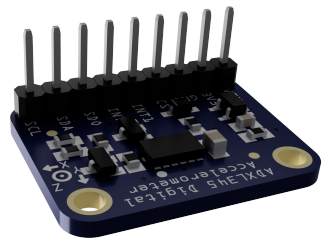


Figure 3.5: Render of the ADXL345 board.

3.2.3 ACS770 Hall-effect current sensor

Hall-effect current sensors are a type of contactless current sensing technology that has gained popularity due to their advantages in terms of size, economic feasibility, low power consumption, high dynamic range, and integrability with standard CMOS technologies [7].

The basic principle of operation for Hall-effect current sensors is the Hall-

effect, which is the generation of a voltage across a conductor when it is placed in a magnetic field perpendicular to the direction of current flow. This voltage is proportional to the strength of the magnetic field and the current flowing through the conductor. By measuring this voltage, one can determine the magnitude of the current flowing through the conductor.

ACS770 is a family of hall-effect current sensors that are often found in high current applications like between a battery and a load in either drones, rovers, or autonomous underwater vehicles. They are slightly more expensive than their widely used counterpart ACS758 which is often used by Chinese circuit board manufacturers. The extra expense is due to their increased performance. One negative about these sensors is the lack of an onboard voltage regulator, this means the sensor needs a separate 4.5V-5V power supply.

3.2.4 Saleae Logic Pro 8

The Saleae Logic Pro 8, is a portable USB logic analyzer with 8 input channels dual-purposed for analog data recording. Some of the important specifications of the device are given in table 3.2

Table 3.2: Saleae key performance attributes.

Features	Performance
Analog sampling rate	50 MSPS
ADC number of bits	12 bit
Analog volts per bit	4.88mV
Analog voltage range	-10V to 10V

The Saleae Logic Pro 8 uses software called Logic 2, an image from this can be found in appendix B. It is a fast and user-friendly interface between the user and the hardware. It does by design lack many of the functions that larger software packages have. These functionalities can be added through extensions either made by the manufacturer or the community.

3.3 Signal processing

Data streams from analog and digital sensors should be organized and analyzed in a way such that useful information can be presented about the system in question. This can be performed in many different ways, and using the right analysis tools is crucial to extract the wanted information. After gathering the data, there is usually some post-processing or cleaning of the data. This is performed either to remove noise, make the data easier to handle, or augment in some way that helps the user gain insight into the system.

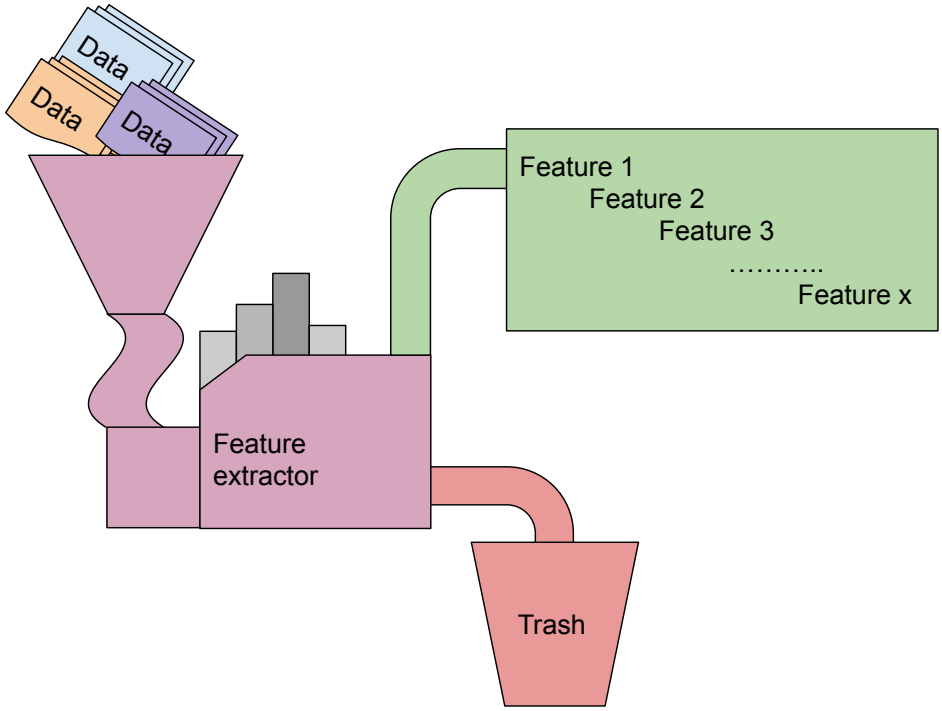


Figure 3.6: The flow of extracting useful knowledge from data. The useful data is extracted from a data stream. The useful data is processed into organized knowledge.

A time domain approach uses the signal, i.e. power, voltage, acceleration, or any other measurable quantity, and plots its value over time. This is useful in a lot of applications; position, temperature, and speed are some examples. A high-frequency signal with multiple layers of amplitudes and frequencies can, however, be very challenging to analyze using a time domain approach. To better character-

ize the fault in a part one can use a frequency domain approach as described for bearings in [6].

A frequency domain approach extracts useful information about the amplitude of different frequencies present in the signal. If a periodic knock or imperfection in for example a ball bearing rolling at a constant speed is measured by an accelerometer, there will be a peak in the frequency domain plot at the frequency of the knock. This frequency will in most cases be possible to understand as a particular kind of fault or reveal information about the origin of the fault [6].

3.3.1 Power spectral density

Power spectral density (PSD) is a measure of the distribution of power with respect to frequency for a time series or signal. In other words, it describes the amount of power present in a signal at each frequency. In the field of mechanical analysis, power spectral density or other frequency-based analysis approaches are often used to analyze the vibration behavior of mechanical systems [10]. The PSD of a signal $x(t)$ can be calculated using the Fourier transform of the autocorrelation function [20].

$$PSD(f) = \mathcal{F}\{R_{xx}(t)\} \tag{3.1}$$

where $\mathcal{F}$ denotes the Fourier transform, f denotes frequency, and the autocorrelation function R_{xx} of the signal $x(t)$ is defined as the average value of the product of the signal at time t the same signal time shifted by τ

$$R_{xx}(\tau) = \mathbb{E}[x(t)x(t+\tau)] \tag{3.2}$$

$\mathbb{E}$ is the expected value operator [20].

One of the applications of power spectral density in mechanical analysis is in

the identification of the natural frequencies of a system. By analyzing the power spectral density of a system's vibrations, it is possible to determine the frequencies at which the system is most likely to vibrate. This is important because the natural frequencies of a system determine its stability and its response to external forces.

Another application of power spectral density in mechanical analysis, more relevant to this thesis, is in the prediction of fatigue failure in mechanical components. By analyzing the power spectral density of a system's vibrations, it is possible to determine what frequencies relate to what kind of failure or fatigue in the system.

3.3.2 Time-domain statistical features

Time-domain statistical features are used to reveal information about a time-domain signal. This can for example be acceleration or motor current as a function of time. By sampling these features over time as a component wears down in use one can investigate whether the feature has a correlation with for example the RUL of the component. The usage of some of these features is shown in [19].

Kurtosis

Kurtosis is a statistical measure that quantifies the shape of a probability distribution or frequency distribution of a data set. It provides information about the tail behavior and the presence of outliers in the data.

The kurtosis of a distribution is defined as the fourth standardized moment, divided by the square of the variance. In simpler terms, it measures the heaviness of the tails of a distribution relative to the normal distribution [2].

$$Ku = \frac{\sum_{i=1}^{N}(x_i - m)^4}{(N-1)\sigma^4} \tag{3.3}$$

Positive kurtosis indicates heavier tails or distribution with more extreme outliers than the normal distribution. This implies that the distribution has more values

in the tails and may exhibit more extreme values than would be expected under a normal distribution. It is often associated with peakedness in the center of the distribution.

Negative kurtosis, on the other hand, indicates lighter tails or distribution with fewer outliers than the normal distribution. This suggests that the distribution has fewer values in the tails and may have a flatter peak compared to a normal distribution.

Crest Factor

Crest factor is a measure that quantifies the peak amplitude of a waveform relative to its average or RMS (Root Mean Square) value. It provides information about the dynamic range or the difference between the highest and average signal levels.

$$CF = \frac{max|x_i|}{\sqrt{\frac{1}{N}\sum_{i=1}^{N} x_i^2}} \tag{3.4}$$

In simple terms, crest factor tells you how much the waveform's peaks exceed its average level. A high crest factor indicates large peak amplitudes relative to the average, meaning there are significant spikes or transients in the signal. A low crest factor suggests smaller peak amplitudes relative to the average, indicating a more consistent or constant waveform [11].

RMS

Root Mean Square (RMS) is a statistical measure that calculates the square root of the average of the squared values in a set of numbers. It provides a way to measure the magnitude or intensity of a signal or a set of values.

$$RMS = \sqrt{\frac{1}{N}\sum_{i=1}^{N} x_i^2} \tag{3.5}$$

In simpler terms, RMS gives you a value that represents the typical or ”average” magnitude of a set of numbers, while taking into account both positive and negative values. It is often used to describe the effective or equivalent value of a varying quantity, such as the voltage or current in an electrical signal.

RMS is widely used in various fields, including physics, engineering, and statistics. It is particularly useful for describing the amplitude or power of periodic or varying signals, as it provides a single value that summarizes their overall magnitude [11].

Shape Factor

The shape factor is a numerical measure that describes the deviation of a probability distribution from a reference shape, offering insights into its asymmetry, peakedness, and tail behavior. It helps assess the unique shape characteristics of the distribution [1].

$$SF = \frac{\sqrt{\frac{1}{N}\sum_{i=1}^{N} x_i^2}}{\frac{1}{N}\sum_{i=1}^{N} |x_i|} \tag{3.6}$$

3.3.3 Motor Current Signature Analysis

Stepping further into the uses of PSD, MCSA uses measurements of the input current in motors to generate a frequency signature. MCSA is based on the recognition that electric motors act as a transducer. The motor efficiently detects small time-dependent motor load variations in the mechanical system and converts them into electrical current signals that flow in the motor current cables [15]. These signals are small in relation to the total current drawn by the motor, but they can be extracted reliably and non-intrusively. After some light processing, the signal can provide indicators of the condition, called signatures, of the motor [15]. The trend of the signatures can be determined over time to give information about the condition of both the motor and load [15]. For large slowly rotating electric motors, the speed of the required data acquisition is relatively low. For smaller fast-spinning

motors like light drone motors, the data has to be sampled at a relatively high frequency not to loose valuable information. As a general rule to avoid frequency folding and corrupt data, one has to sample faster than the Nyquist frequency of the signal. The Nyquist frequency is defined as $0.5cycles/sample$ in other words total sample rate in Hz must be twice the frequency of the maximum frequency of your signal, or higher [24].

3.3.4 Harmonics

Usually, harmonics are explained by multiples of some given frequency. When doing MCSA on a motor there are two types of harmonics, mechanical harmonics and electrical harmonics. The mechanical interpretation of the frequency of rotation follows intuition, the electrical is however somewhat different. In an ER-BLDC motor, an electrical rotation is interpreted as one full cycle of electrical commutation in any one phase. To get the frequency of electrical rotation there is a simple formula relating the electrical RPM to the number of poles and the mechanical RPM this formula is shown in equation 3.7, a variation on the formula from [18].

$$\text{RPM}_{\text{mechanical}} * \frac{\text{Number of poles}}{2} = \text{RPM}_{\text{electrical}} \tag{3.7}$$

Harmonics are often used when investigating the frequency spectrum of a signal as they will naturally have higher peaks of power than adjacent frequencies. In the case of the electrical frequency of a BLDC motor, the harmonics will appear at known positions that can be calculated from the number of pole pairs and the mechanical frequency. This makes harmonics potential feature candidates to monitor when looking for signature changes from a healthy motor to a damaged one.

3.3.5 Bearing Fault Frequencies

In ball bearings, there are four common bearing fault frequencies that are often encountered, these are detailed in [8] and referenced below with an explanation of parameters:

- Ball defect frequency (f_{BD})
- Outer race defect frequency (f_{OD})
- Inner race defect frequency (f_{ID})
- Cage defect frequency (f_{CD})

All of these frequencies can be described by the equations in 3.3.5.

$$f_{\mathrm{BD}} = \frac{\mathrm{PD}}{2\mathrm{BD}} f_{rm} \left(1 - \left(\frac{\mathrm{BD}}{\mathrm{PD}}\right)^2 \cos^2 \phi\right) \tag{3.8}$$

$$f_{\mathrm{OD}} = \frac{n}{2} f_{rm} \left(1 - \frac{\mathrm{BD}}{\mathrm{PD}} \cos \phi\right) \tag{3.9}$$

$$f_{\mathrm{ID}} = \frac{n}{2} f_{rm} \left(1 + \frac{\mathrm{BD}}{\mathrm{PD}} \cos \phi\right) \tag{3.10}$$

$$f_{\mathrm{CD}} = \frac{1}{2} f_{rm} \left(1 - \frac{\mathrm{BD}}{\mathrm{PD}} \cos \phi\right) \tag{3.11}$$

Where the parameters are as follows for the 696 EZO bearing used in the MN505-s 260KV.

- Shaft Rotational Speed (f_{rm}) = Not constant.
- Ball Diameter (BD) = 6 mm
- Pitch Diameter (PD) = 15 mm
- Contact Angle (ϕ) = 33,6 degrees
- Total Number of Balls (n) = 7 balls

3.3.6 Norming data

Particularly when working with multi-axis or multi-phase data, working with the norm of all the components can be easier than working with the data component for

component. This greatly reduces the size of the data set but might hide interesting features. The norm is given by equation 3.3.6.

$$||\mathbf{x}|| = \sqrt{x_1^2 + x_2^2 + ... + x_n^2} \tag{3.12}$$

4

Aviant's drone system

This section is loosely based on previous work done in a project thesis on a different part of the same drone system. The drone platform currently in use by Aviant is called Notus. The drone frame is of the model Baby Shark 260 VTOL produced by FoxTechFPV, which also supplies the ER-BLDCs and ESCs. The control electronics, sensors, and actuators on any given Notus model are provided by and/or customized in-house at Aviant. A rendering of the drone and its components is presented in figure 4.1.

4.1 Airframe

The airframe consists of a fuselage with two detachable wings with their own detachable VTOL arms. These arms have screw connectors in the rear for connecting an A-tail. The airframe is a quite standard VTOL, the wings are equipped with ailerons and VTOL-arms with top motors for vertical takeoff or landing. Where the airframe differs from the convention is in its A-tail configuration. It combines the control force of the more traditional rudder and elevators into two symmetrical ruddervators, the combination of actuator force on these control surfaces are now responsible for both pitch, θ, and yaw, ϕ. The concept of an A-tail is similar to that of a V-tail, but upside down and with spacing so that the tail can be behind a

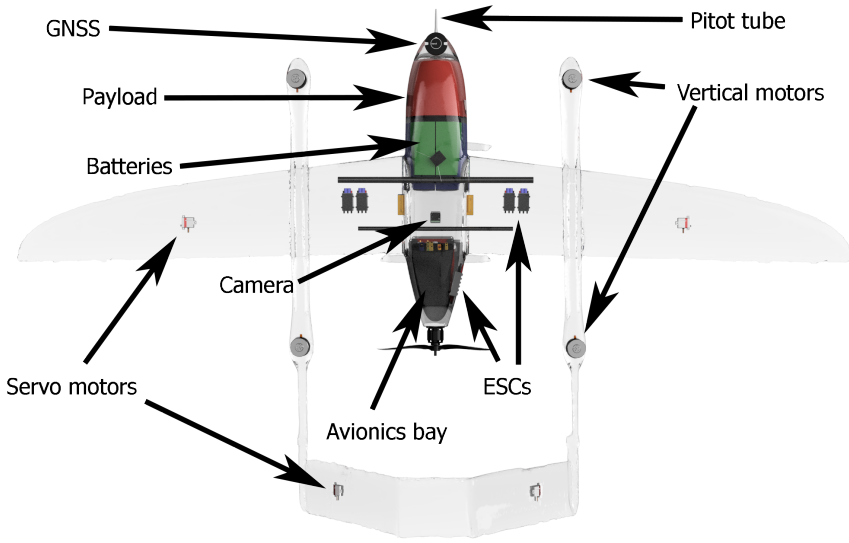


Figure 4.1: Notus.

pusher motor. V-tail configuration in an unorthodox drone form factor is described in [3], the control laws for A-tails and V-tails are very similar.

Table 4.1: Physical features of Notus.

Physical characteristic	Value
Wingspan	250 cm
Length	144 cm
Height	55 cm
Wing area	75 sq. dm
Airframe mass	2.7 kg
Empty mass	5.6 kg
Material	Composite
Maximum takeoff weight	14 kg

4.2 Motor and ESC

The top motors and top ESCs are made by T-Motor. The motor is a navigator type, MN505-s 260KV, where the KV number refers to how many revolutions per minute the motor will do at a specific voltage at max throttle. For example if the

Table 4.2: Performance characteristics of Notus.

Performance characteristic	Value
Max altitude	10,000 ft
Endurance	105 min
Range	145 km
Cruise speed	23 m/s
Max cruise speed	27 m/s
Stall speed	15 m/s
Rate of climb	5 m/s
Rate of descent	5 m/s

motor is run at $50\,\mathrm{V}$ we have

$$50\,\mathrm{V} \cdot 260\,\mathrm{RPM/V} = 13\,000\,\mathrm{RPM} \tag{4.1}$$

Though this value is dependent on the pitch of the propeller in use, from the equation we also see that the max RPM also changes with voltage. This means that a fully charged battery will be able to spin the motor faster than a more discharged battery, this also follows intuition. The 260KV variant has no public datasheet as it is not a standard T-motor product, but rather a special production run for FoxTechFPV. The ESC is model FLAME 60A rated for 60A continuous operation and a 10-second peak of 80A.

4.3 Signaling and electronics around the motor and ESC

In the Notus platform, there are 4 motors, two per arm, this makes the platform function as a quadcopter when in multirotor flight. The electrical circuit and the signal paths are illustrated in 4.2. When the flight controller has a new reference signal it is sent as a PWM signal to the correct ESC through the main outputs of the onboard flight controller. Power is supplied directly from the battery through a current sensor. This sensor keeps track of the total consumed power of the system, this is important both for range estimation and to minimize the risk of damaging the battery. The signaling is identical for all the motors. Normally the PWM signals are generated by the flight controller, but to streamline the process of data

generation and harvesting while testing, the PWM signal is generated by the raspberry pi, which also controls the rest of the test rig.

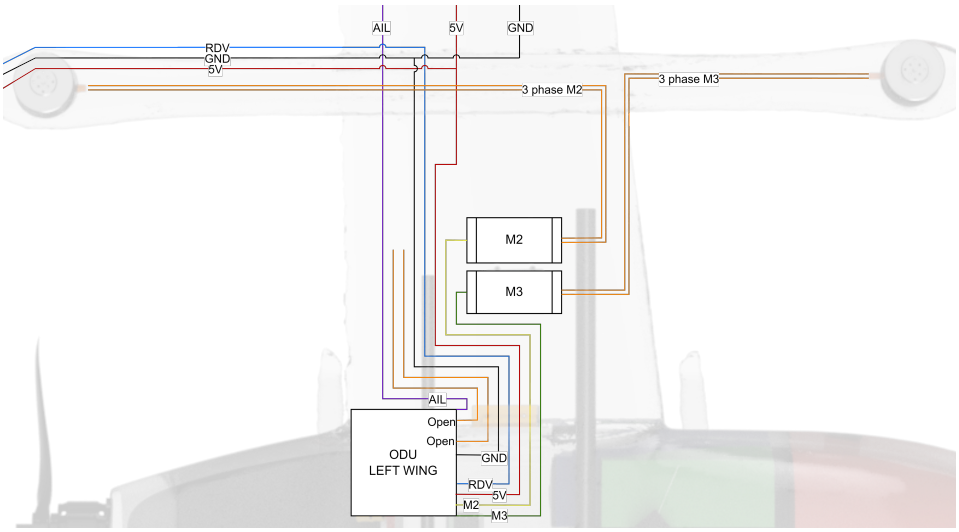


Figure 4.2: Illustration of left side signal wiring from avionics bay to wing, VTOL-arm, and tail, including phase power from the top ESCs to the top motors.

4.4 Failure modes of Notus

When discussing the top motor system and how to apply new detection methods and thus introduce complexity, it is important to know the consequences of the system failing. Notus, being a VTOL drone, has the advantage of using the quad chute maneuver if control is lost while being in fixed-wing mode. In a quad chute, the drone will activate all the top motors while in forward flight, thus the quadcopter control system takes over. The drone is no longer dependent on the wings or its control surfaces. It does however not have the option of doing an emergency transition the other way, that is, from multirotor to fixed-wing flight. This means that a top motor failure will lead to the drone being uncontrollable, thus it will have an unsafe failure.

To mitigate the risk of having an unsafe multirotor failure multiple actions can be taken. One action is to make the system over-actuated. The standard Notus has 4 top motors, increasing this to 8 would make it possible to make an emergency

landing even if one motor were to fail. This solution would however introduce much complexity and weight, it would be expensive to add more motors, and also the maintenance of more motors and propellers would make this solution sub-optimal. Another solution would be a self-deploying parachute, this way the drone always has a safe failure mode even if a top motor fails to generate enough thrust. A parachute also adds complexity and weight, but could be a completely separate system not relying on for example the battery of the drone to function.

4.5 Maintenance of Notus

The starting point for making a new maintenance strategy of the Notus platform is shown in figure 4.3 where the safety of flight is highly dependent on two things. Firstly the operations crew, pilots, and commanders need to do checks and monitor the drone's health based on their training and intuition. Secondly revealing potential failure points while doing a deeper inspection in the drone lab. This deeper inspection should uncover any fault given that the drone does not crash until the inspection counter runs out. Even though this inspection is designed to uncover all critical faults, there is a chance that the personnel does not uncover all faults. To decrease the time in flight with an unknown fault the inspection timer must be short enough to where the chance of crashing before the next inspection is adequately low. Thus the technicians and engineers at the lab in theory have multiple chances to discover the same fault, further increasing the chance of finding the fault before it results in a crash. Currently, this conservative calendar-based maintenance is the main safety barrier for human errors during an inspection.

This approach is naive at best and very costly at worst. Inspecting and maintaining the fleet when it is not strictly necessary and relying solely on the humans operating and developing the drones to uncover the faults in such a complex system is likely not the optimal solution.

A more ideal way of maintaining the fleet would be for the drones to find most faults by themselves. A possible new maintenance strategy is shown in figure 4.4. The better the CM system, the more of the flow in the diagram will go through the CM step. This way humans can make subjective decisions on criticality and safety,

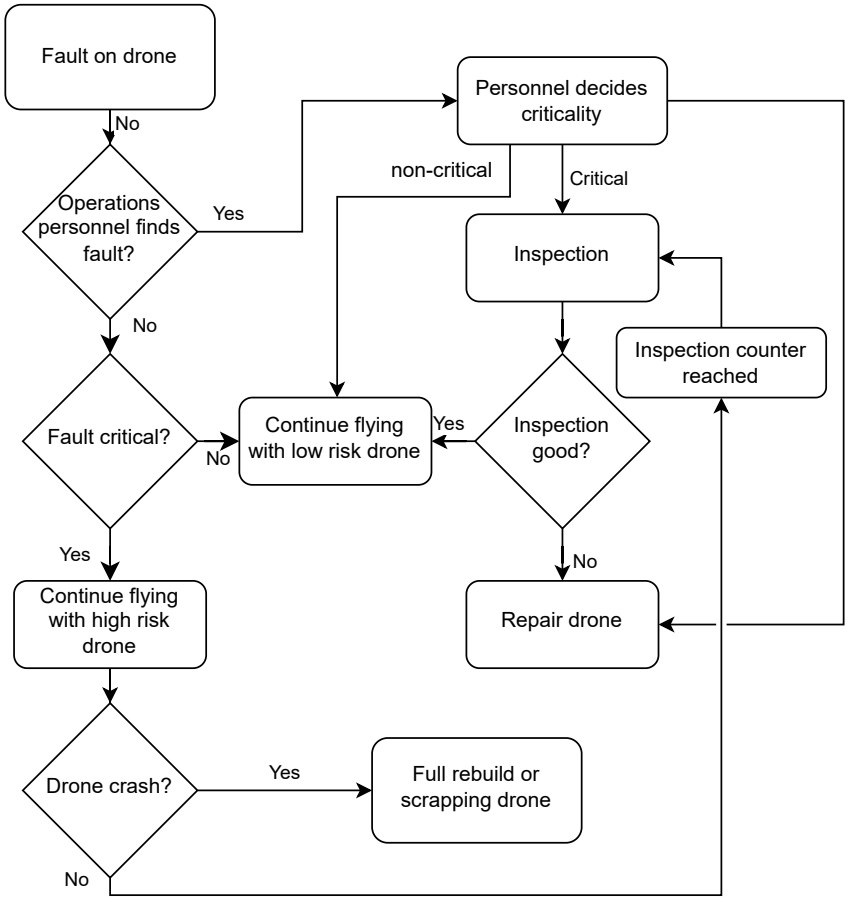


Figure 4.3: Logistics before CM.

while computers can analyze data. Analyzing the data in a correct manner should give the drone and the operators a good insight into its health, performance, and its components RUL.

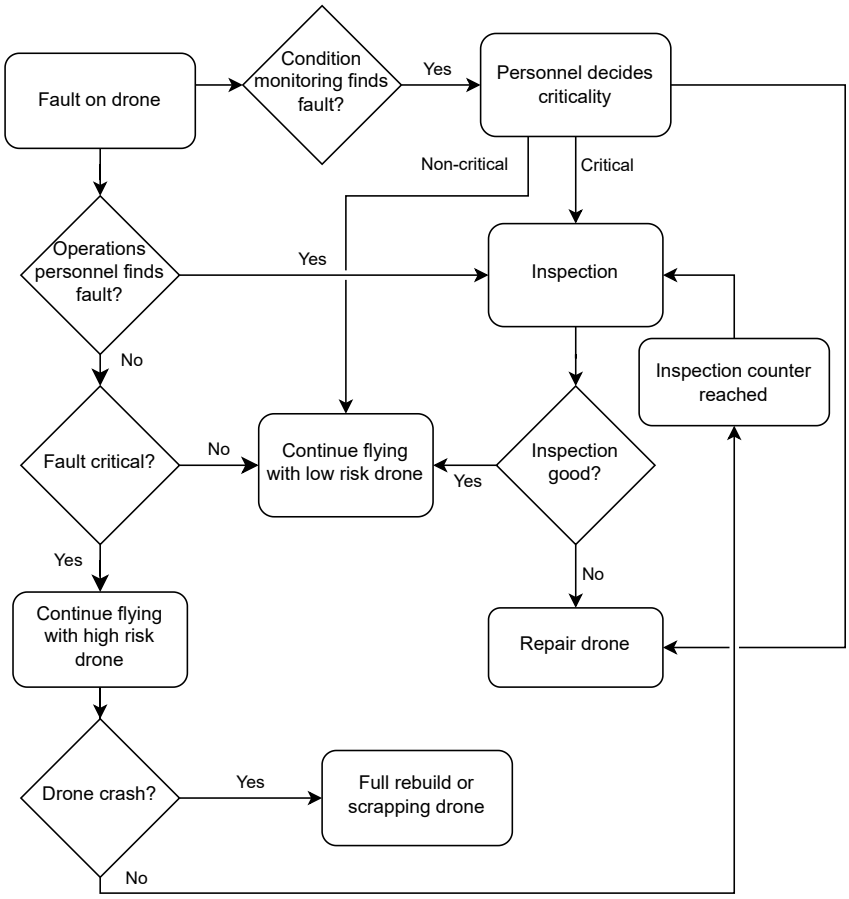


Figure 4.4: Logistics after CM.

5

Method

5.1 System composition

5.1.1 Wing

The wing is a standard drone wing made of aramid fiber, foam core, and balsa wood ribs and spares. Compared to carbon fiber, aramid gives better abrasion resistance, less chance of cracking, and higher yield strength (resistance to plastic deformation). On the wing there is an aileron, this control surface is responsible for rolling the aircraft. The control surface is hinged to the rest of the wing in a compliant joint made from the same piece of aramid as the rest of the wing.

5.1.2 VTOL arm

The VTOL arm is made of two 1mm thick carbon fiber halves that are glued together with a semi-flexible epoxy-like glue and balsa wood ribs and spares spaced throughout the arm. On the rear end of the arm, there is an extending piece made from 25mm x 1.5mm carbon fiber tubing where a screw connection is fastened with rivets for easy assembly and disassembly of the A-tail. The motor mounts are made by embedding balsa wood into the carbon fiber layup and curing it inside. Holes are drilled under every motor mount to allow for access to the mounting

hardware of the motor. This hole also allows for the mounting of some of the sensors in use in this thesis.

5.2 Test environment

Testing was done on a custom-built test rig, it was purpose-made for testing different sensor, motor, and propeller combinations in a safe manner either at the location or remotely. The main body of the test rig is constructed in wood as it is easily available and quick to make frames of. On the main body, there is space for mounting a wing in the same way it would be mounted on a drone body, using carbon tubes and special thumb nuts called wing nuts.

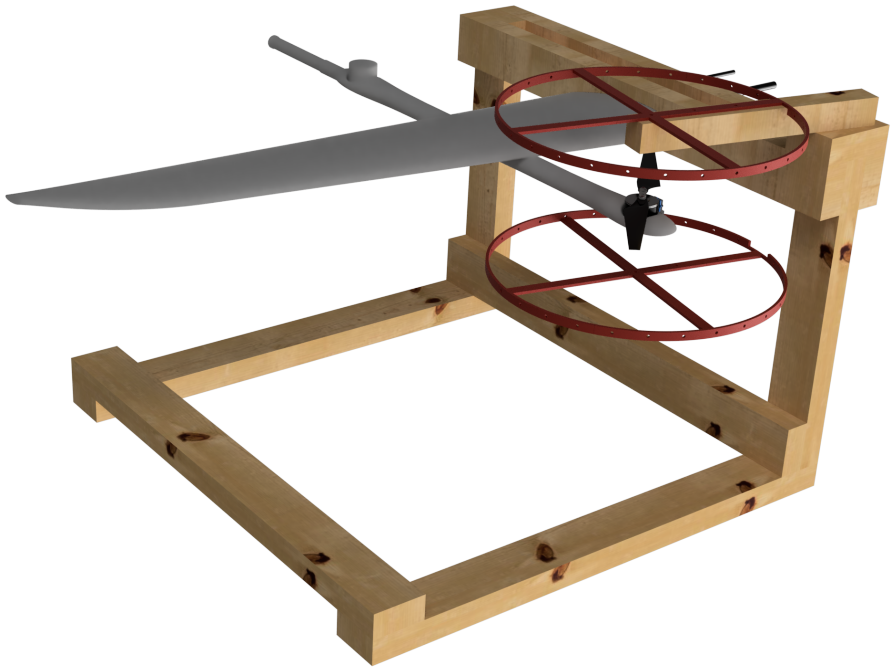


Figure 5.1: Render of the 3d model of the test stand. The stand was built in wood.

5.2.1 Power supply

In operation the drones use a lithium battery for power, such batteries can store a lot of energy and also release the energy at a very rapid rate. Finding a power supply (PSU) to deliver the same amount of power at a strict budget is challenging. The solution was to combine power supplies that were not in use at the office. The required specs were dictated by the motor, which is rated for 1300 watts at 12S (44.4V). So the target for amperage was calculated, shown in eq. 5.2.1.

$$1300\,\mathrm{W}/44.4\,\mathrm{V} \approx 30\,\mathrm{A} \tag{5.1}$$

The final configuration was to combine two In-Win units at 48V 12.5A each and connect them in series to get a combined output of 25A at 48V. This increases the rotational speed of the motors slightly, but since the 44.4V is the nominal voltage of the battery this configuration is a good midway between the nominal and the peak 50.4V at full charge. This is a general property of Li-Po and Li-ion batteries that change somewhat from one chemistry to another, usually one operates with a nominal voltage of 3.6V to 3.7V per cell for Li-Po and Li-ion [23]. FoxTech, the manufacturer of the batteries normally in use in the Notus drone, and other suppliers often work with 3.7V nominal voltage, for the sake of comparison this is what the calculations in this thesis will be based on. For completeness, the calculation of a 12S semisolid state FoxTech Diamond Li-ion battery is shown in eq. 5.2.1

$$12\,\mathrm{cells} \cdot 3.7\,\mathrm{V}/\mathrm{cell} = 44.4\,\mathrm{V} \tag{5.2}$$

For safety, the power supplies were not connected directly to the power outlet but were wired through a C14 receptacle and fuse combination with a 10A glass

fuse, as well as an I2C-controlled 230V relay for remote restart of the test stand.

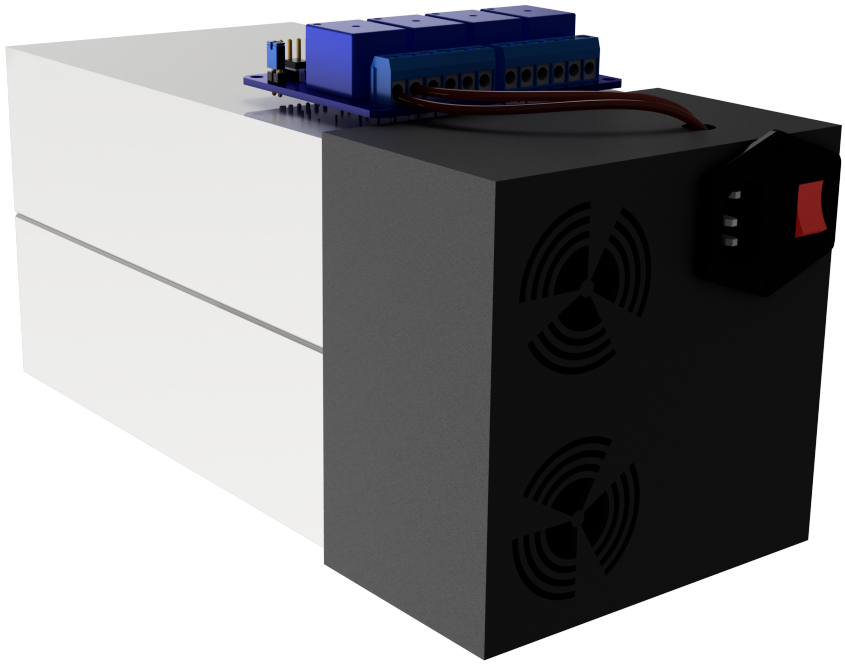


Figure 5.2: Render of the Power supply with 4 channel relay on top, the holes in the illustration are for the two fans on the units.

5.2.2 3d-modelling and printing

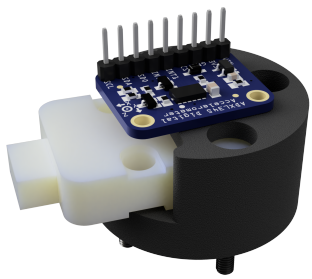
Modeling all the parts needed were done in Fusion360, and printing was done on fused deposition modeling (FDM) printers. All parts were printed in PLA as there was little to no requirement for heat deflection. To optimize for the FDM production method, the parts were modelled to be printed without support structures as far as possible.

5.2.3 Sensors and mounts

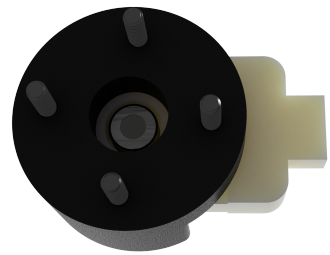
The sensors and their measuring principles are presented in chapter 2, this section will cover mounting and positioning.

ADXL345 and MLX90614

For acceleration the best placement is a stiff mounting close to where the acceleration in question is happening, this results in mounting close to the motor. This is to get a less damped response in the measurements. To achieve this the ADXL345 is mounted using epoxy on a 3d-printed component that is again mounted with epoxy to the motor mounting plate of the VTOL arm. Since the contact-less temperature sensor (MLX90614) should measure the temperature as close as possible to the stator and windings of the motor, it too has to be mounted on the underside of the motor. The final design of the mount is illustrated in figure 5.3.



(a) Top view of the accelerometer and contact-less temperature sensor package.



(b) Bottom view of the accelerometer and contact-less temperature sensor package.

Figure 5.3: Accelerometer and contact-less temperature sensor.

Current sensors

The current sensors are connected to each of the phase current wires of the ESC and held to the wing with double-sided tape. This was done to have a quick and easy mounting method where it was easy to change the sensors if there was ever a situation where one needed to remove one or more of them. The current sensors in use are partly supplied by Mauch Electronics. The sensors are originally based on an ACS758 unidirectional hall effect current sensor. The boards had to be modified to measure current both ways as half of the driving current is negative. The circuit board was modified with a bidirectional ACS770 chip with a max current of 100A.

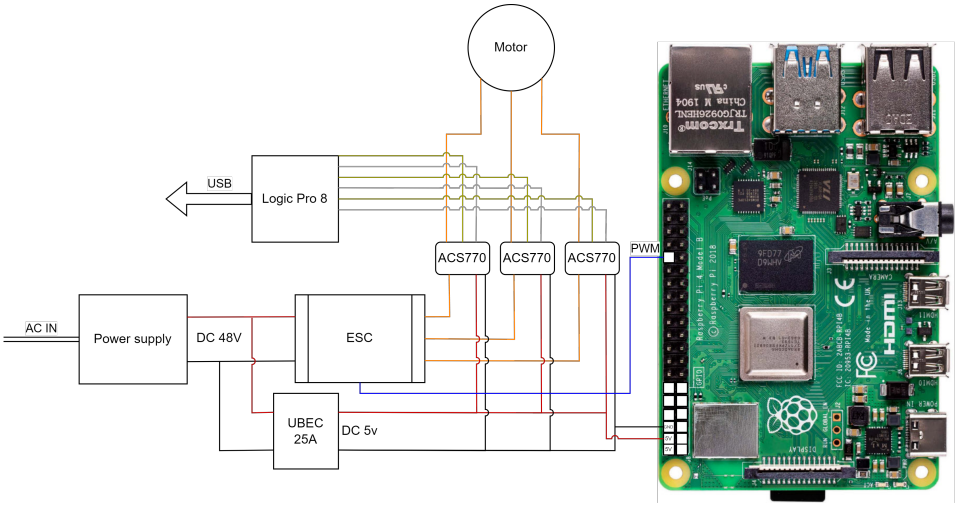


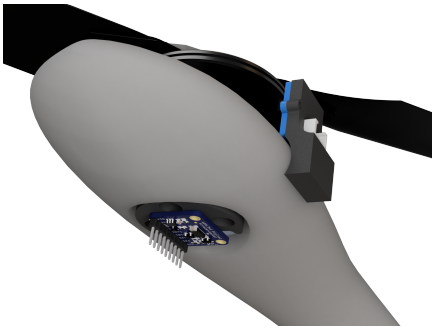
Figure 5.4: The wiring diagram for power supply, motor controller, phase current measurement, and motor.

Tachometer

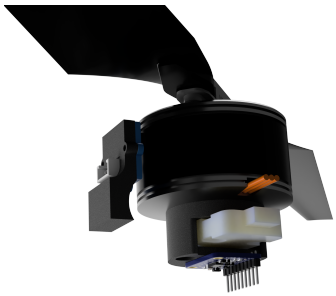
The tachometer on the test rig is made from an RPR-200 reflective photosensor with carrier board screwed to a 3d-printed holder which is glued to the VTOL arm. The sensor is pointed directly at the rotor housing on the motor, and a small reflective piece of tape is stuck to the outside of the rotor. This way every time the reflective piece rotates past the sensor, a data pin goes from low to high on the rpi and is registered as a rotation. With some simple code, this can be translated to RPM.

The equation for converting the reflector signal to RPM is given in equation 5.3, note that there are 2 changes in state (on/off) every revolution, therefor dividing by 2.

$$\frac{SensorStateChanges}{2 * 1s} * 60s/min = RevolutionsPerMinute \tag{5.3}$$



(a) Component stack on VTOL-arm, to the right is the mounting of the reflective sensor, and on the underside of the motor the mount for the temperature sensor and accelerometer is shown. cables are not modeled here for simplicity.



(b) Component stack seen with invisible VTOL-arm, to get a better view of the temperature sensor (white in the picture), the camera was pointed from the back and forward.

Figure 5.5: Component stack on the front motor of the VTOL-arm.

5.2.4 Calibrating the Logic pro 8

Using the ACS770LCB-100B-PFF-T the measured voltage on the current output on the circuit will only have positive values even when back EMF is present. Wanting to also measure back EMF negative values had to be enabled. Since changing the hardware was deemed complicated the solution was tampering with the calibration of the Saleae. A user by the name Marcus10110 on github has written a script to edit the calibration files of the Saleae. The file can as of May 5th 2023 be found here [Marcus10110 Saleae calibrator] [Marcus10110]. The tool is written in Python and runs in the console. When running the script the user gives the location of the calibration file and a set of gains and offsets for all the channels. The script then creates a backup of the original calibration and implements the new values in the calibration file.

The final calibration for the Saleae was implemented with this command

```
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.

c:\User> python saleae_ca_editor.py "c:\path\to\calibration\file.cal"
--ch0-offset 4.99 --ch1-offset 4.845 --ch2-offset 4.825 --ch0-gain 3.185877
-ch1-gain 3.129401 --ch2-gain 3.134064
```

The goal of the calibration was to make the voltage signal of channels 0, 1, and 2 in the Logic 2 software represent the real-life amps flowing through the phase cables of the motor. The limitation was however that the software is limited to only display up to +-10V. This meant if a one-to-one representation of displayed voltage and real-life amps were to be implemented in the calibration the peaks would go out of view in the software. This is unwanted behavior as analysis of the signal directly in Logic 2 would be impossible. To solve the issue a 1 to 10 factor was chosen. Such that 1V in the software equal 10A flowing in the phase cable.

To find the right gain and offset the phase current was measured when zero speed was commanded. The median of this signal was then offset to zero. After this a gain of ≈ 3 was added to the gain and, this way the theoretical maximum of the output will be $\approx$ 6.2V which will indicate 100A of current through the cable. Having not used the full range of the software this leaves room for spikes higher than 100A. The data sheet of the ACS770LCB-100B-PFF-T describes that the rating of 100A is only based on the max operating temperature, this way the chip can handle spikes of over 100A if there is temperature headroom. The final scaling factor to go from displayed voltage to real-life amperage is given in 5.4.

$$\text{[Scale from data sheet]} \cdot \text{[Gain]} = \text{[New scaling]} \tag{5.4}$$

$$20m\text{V/A} \cdot 3.15 = 63m\text{V/A} \tag{5.5}$$

This example holds in the general case, the scaling factors were tweaked somewhat to make the zero level correct in the Logic 2 software

5.2.5 Remote and emergency stop functionality

To make the testing safer when not overlooking the test equipment a number of key parameters of the test equipment have limit values that trigger the relays and turn off power to the motor and ESC. The values used are temperature, rpm, and acceleration. The increase over a set maximum of any of these parameters would trigger a full stop only leaving the raspberry pi and the sensors connected on to

further monitor and possibly restart the system. The functionality of the relays are not limited to safety as a common issue when testing was for the motor to de-synchronize, that is when the ESC is controlling the phase currents in such a manner that the physical system cannot follow. From experience with the combination of T-motor flame 60 ESC and T-motor MN505-S, this commonly happens when the ESC's input increases in value too fast, or when the ESC works beyond its rated 60A. This would for example happen with too large of a propeller.

5.2.6 Safety cage

The safety cage is made as a last line of defense if the propeller was to fail. If a person was in close proximity to the test rig while the motor was running with a bad propeller, the safety cage is designed to catch large pieces of the propeller coming off. Smaller pieces with very little mass are not of concern as these pieces would lose their impact energy very quickly. The safety cage is made by 3d-printing a circular frame that chicken wire can be suspended between, the frame and chicken wire should interfere little with the airflow around the propeller, while giving adequate protection.

5.2.7 Raspberry pi PWM note

The raspberry pi does not use its real-time clock for PWM signaling out of the box. This makes the timing of the PWM signals imprecise to such a degree that it clearly affects the behavior of the motor. To test this, a test code was written to hold a servo motor still with a constant PWM reference signal. Before this issue was fixed, the test showed that the servo could not be held steady, but rather vibrated in a random manner when given a "constant" servo signal (the code was supposed to give constant PWM, but the signal clearly was not). This makes sense since the raspberry pi emulates a HW clock using software when not explicitly told to use the HW clock for PWM signaling. This would significantly affect the data if not fixed. The issue was fixed using a service connecting the gpio outputs of the raspberry pi with the hardware clock on the chip. The service, GPIOD [TronicsBench], fixed the issue completely. After implementation, the pi has no issue holding the servo completely still.

5.3 Testing methodology

5.3.1 Standard testing criterion

To make the test conditions similar from one test to another, temperature and RPM was measured when the tests were performed. The test was done indoors and the temperature in the room was not controlled. All data was gathered when the motor was between 35C and 50C. The temperature was measured with a contactless sensor pointed at the lower bearing. The propeller was torqued to spec for every test to guarantee the propellers were similarly mounted. Note that testing revealed weaknesses in these criterion, this will be discussed in section 6.3.3.

5.3.2 Trending and comparison tests

For the general test procedures, there were mainly two types of tests conducted. Trending tests were run one after another changing one parameter or modification for every test. For example off-center weight on the propeller was lightly increased test after test. From these results trending graphs of different features could be plotted. Comparison tests were conducted on different confidurations, for example one propeller to the next or with different motors. Features were extracted and compared, and plots of the PSD was made to compare the two configurations. The plots and changes will be explained in more detail in chapter 6.

5.3.3 Method validation test

To validate that the setup works as intended and can be used to isolate signatures in the motor current frequency spectrum the motor and propeller are first mounted properly to create a baseline. The baseline measured for 3 different propellers, the propellers in use are listed in table 5.1. After a baseline is measured with every propeller, the propellers were unbalanced in a reversible manner by adding off-center weight, the weight of choice is tape as it is easy to add and remove. The tape is measured on a scale with a 0.01g sensitivity for logging after every test, the tape is also kept at roughly the same size and position from one test to another. Approximate position for the tape is marked in figure 5.6. Making the weight and position of the tape similar on different tests on the same propeller means the tests

could be repeated if an error is discovered at a later time.

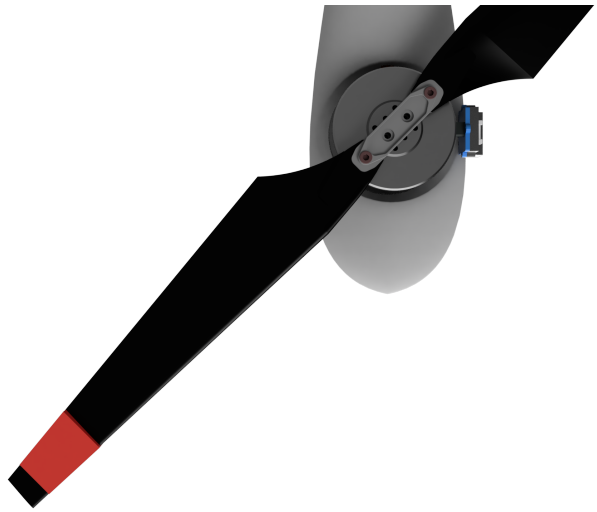


Figure 5.6: Illustration of a drone propeller. The red field in this illustration is where the tape is fastened in the tests for testing up to 1g.

Table 5.1: Table of propellers used in the thesis.

Name	dimensions	type
FoxTech	16x5.5in	Normal
Mad	15.2x5in	Folding
GEMFAN X CLASS	13x10in	Triblade

5.3.4 Temperature testing

While testing a mistake was discovered. The original idea behind measuring temperature was that it was a so-called ”nice to have”. Maybe there would be some small changes in the graphs with varying temperatures. The reality was that temperature had a huge and very much measurable impact. The first set of trend investigating tests, shown in section 6.3.1, was showing no trending at all. Controlling all other variables the graphs looked like the signatures bounced around randomly. After this test, it was deemed necessary to investigate the impact of temperature deeper. A temperature test was done with the same propeller for all the data samples. First, the propeller was spun at high RPM to get the motor warm. Then the

motor was controlled by a steady 1250PWM signal and samples were taken at different temperature intervals as the motor was cooling down. Finding this flaw in methodology was key to the project as a whole. The temperature test is presented in 6.3.3.

5.3.5 Motor damage for testing

Disassembly of motor

To damage the bearings in a controlled way the bearings had to be removed from the stator assembly of the motor. To remove the bearings without accidentally damaging them a bearing puller was made from two Allen keys and a pair of locking pliers. The tools are shown in figure 5.7. When the bearings were removed they could be inspected properly, damaged and put back in the motor.

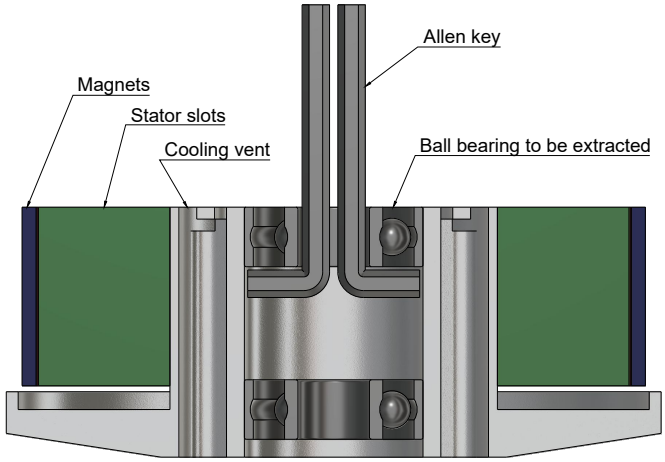


Figure 5.7: Cross-sectional view of the bearing extraction with Allen keys to visualize the extraction process. As the Allen keys are pulled up to extract the bearing.

Damaging the bearing

The goal was to damage the bearing in a way where the inner and/or outer race has a pit or dent that can be felt when spinning the motor by hand.

The first attempt was to damage the bearing using a hammer and a 6mm axle

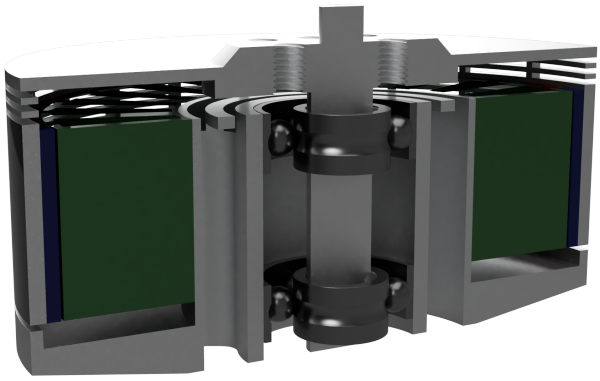


Figure 5.8: Cross-sectional view of the motor with the rotor in place.

in the center of the bearing. The axle is there to prevent the bearing from deforming when it is hit. This approach ended in a crushed bearing shown in figure 5.9.

Figure 5.9: First attempt to damage a bearing in a controlled way. The end result was a crushed bearing.

For the second attempt, the bearing was first heated with a blowtorch so that the bearing lost its hardening from the factory. After the heating cycle, the bearing was washed in isopropyl alcohol to remove burnt grease. When the bearing was unhardened and without grease it was once again hit with a hammer on one side with a 6mm axle in the center. The bearing is now in general running with more

friction and has a substantial dent in the inner and outer race. This can be felt when turning the bearing by hand. The now blackened bearing is depicted in figure 5.10.

Figure 5.10: Bearing after damaging by blow torch and hitting with a hammer. It still has its shape but is severely blackened from the heating and burning of grease.

The bearing damage was done on another motor so that there are two motors that can be tested one after another. The press-fit of the bearing into the stator housing was done by lubricating the housing with grease and pressing with a vice. Figure 5.11 we see the underside of the motor with a now damaged bearing.

Figure 5.11: Under side of the MN505-S where the lower bearing has been replaced with a damaged bearing. One can see that the bearing is a damaged sample by the charring.

5.4 Code

The code is divided into two main projects, the data logging and control of the test stand, and the data processing and analysis running locally on a computer. This section covers some of the coding done during the project.

5.4.1 Test stand code

The code running on the raspberry pi connected to the test stand has multiple drivers and test code, as well as some code to make it easier to do the tests. An illustration of the code architecture on the raspberry pi is shown in figure 5.12. Drivers had to be written for many of the sensors, and even for the sensors that already had drivers, there was some lacking functionality that had to be programmed to make it work. The drivers were then used for control of the hardware using Python threading to manage controlling the motor while also measuring using the sensors. The sensor data was also input to a watchdog that made sure safety thresholds were not breached.

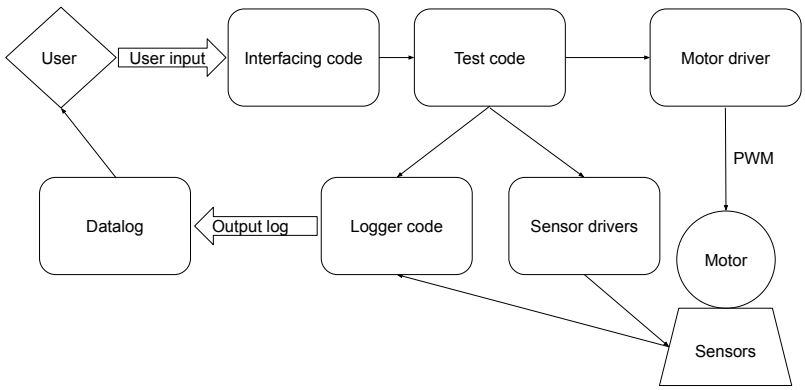


Figure 5.12: Architecture of code running on raspberry pi

5.4.2 Data processing and plotting

Generally, the data is input as a .csv file and transformed into a pandas data frame. All the data processing is done using the data frame and presented as a plot or a value.

To calculate the PSD of the data a Python library called endaq was used. This is an open-source engineering data acquisition library, that among other things has an easy-to-use PSD function that uses the `scipy.signal.welch()` function to compute the PSD. Welch's method is a transform that splits the time-domain signal into smaller overlapping frames, computes the fast Fourier transform (FFT) on every segment, and computes the average of the FFT transformed segments. One can also control the binning of frequencies in the `endaq.calc.psd()` function, this will average the signals in a set interval into one point in the final data set. For example, using a bin width of 2Hz will for example average the power of all the frequencies in the interval 1Hz to 3Hz into the power of 2Hz. Depending on the type of window and the size of the overlap one can control the smoothing and de-noising characteristics of Welch's method. By standard the `scipy.signal.welch()` uses a Hann window and 50% overlap.

The time-domain analyses are coded from scratch to match the formulas given in section 3.3.2. These functions could be optimized in use as they have some common calculations that could be done once for all the functions instead of once per function. That is RMS is a function in itself, and is also used in for example crest factor, caching the RMS would make the code faster when doing both analyses.

5.4.3 Local minimum and maximum finding algorithm

As part of this thesis, an algorithm was needed to find the relative power of the harmonics of the signal's power spectral density. It was deemed faster to make and implement an algorithm for the specific needs in this thesis rather than finding and modifying an existing one. This subsection describes the problem and the development of the algorithm.

Since the RPM on the test stand is only accurate down to a resolution of

30RPM, the exact point of the harmonics will be impossible to know beforehand. This makes it challenging to plot the trend of the harmonics over time since the user will have to manage the data and look at the plots to find the precise harmonics of the plot. Having this much manual labor in a process that ideally is going to run on an autonomous drone is unwanted.

The goal was to input a PSD data set with an approximation of the harmonic frequencies and output a labeled graph where the algorithm has picked the correct power of all the wanted harmonics. This was however not trivial. There were a couple of large problems that needed to be solved for the algorithm to work reliably. The first big problem was how to pick the correct value from a set. Since the algorithm should be able to find both peaks and valleys, one approach is to find a mean value as a comparison to the highest and the lowest value in the subset. When having linear signed data one can use this approach. Doing this for logarithmic data would result in always picking the high point. Instead, the solution is to make the comparison point a logarithmic mean (table 5.2), and calculate the number of multiples to get to that point from the top and bottom point in the subset. Another alternative to this is using the median, but this only works if there is substantial resolution in the PSD data.

$$10^{\text{mean}[log(i_1),log(i_2)...log(i_3)]} \tag{5.6}$$

Table 5.2: Table showing the difference between a linear and a logarithmic mean.

Type of mean	Normal mean	Log mean
Before operation	[0.1,0.01,0.001]	[0.1,0.01,0.001]
First step	0.1+0.01+0.001	log0.1+log0.01+log0.001
Second step	0.111/5	$-(1+2+3)/5 = 10^{-6/5}$
Value	= 0.0222	= 0.063

When the algorithm could pick the correct top or bottom value from a subset the next problem is choosing the subset. As the RPM is somewhat known the start point was to search from the RPM divided by 60 and multiplied by the har-

monic. This way the unit becomes Hz. From experimentation, one way to get better estimation was having two subsets of points to search. One wider subset for calculating the mean/median and one smaller subset in the middle of this subset that had candidate points for top or bottom points. This way one can exploit the known lower limit of the harmonic from the measured RPM. After testing the final equations were as presented in equations 5.7 to 5.11. These equations are not made to fit all situations, but they worked very well for this specific setup.

$$\text{nSteps} = (3 + (H_n/2 * \text{bin})) * 3 \tag{5.7}$$

$$\text{kStart} = -(\text{nSteps})//4) \tag{5.8}$$

$$\text{kEnd} = 3 * (\text{nSteps})//4 + (\text{nSteps} + \text{kStart} - 3 * (\text{nSteps})//4) \tag{5.9}$$

$$\text{pStart} = \text{RPM}_{measured} \tag{5.10}$$

$$\text{pEnd} = 2 * \text{nSteps}//4 \tag{5.11}$$

Where nSteps is the total number of points to search, H_n is the nth harmonic and bin is the bin width. Small k is a list that defines the start and end of the mean/median search area and small p is a list that defines eligible points for the high or low point. The search area thus has the real harmonic approx centered in the search area, as the measured harmonic is a bit lower than the real harmonic. And the points eligible for top or bottom are members of a smaller subset starting at the measured harmonic and stretching to 2/4th of the mean/median interval. The "$//$" indicates hole number division, in the coded calculations care is taken to always have the length of k be equal to nSteps even though hole number division is used.

As an example of the algorithm in use in figure 5.13. The figure shows eligible points that the algorithm picks out, the mean/median, and the point finally chosen as the point of interest. Figure 5.14 shows the algorithm working on the electrical harmonics.

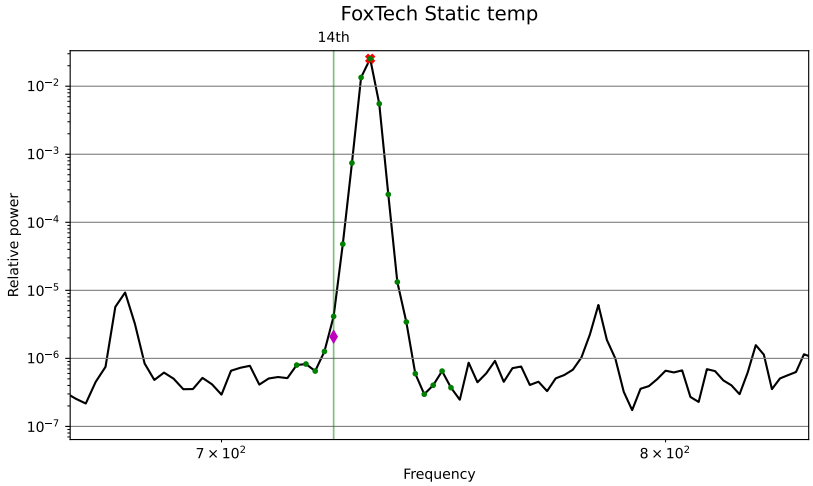


Figure 5.13: Harmonic min/max finding algorithm visualized on a normalized three-phase motor current. The Green dots are the interval k and the green line is the 14th harmonic (1st Electric harmonic), the purple diamond is the median of the values in k and the red cross is the value the algorithm has deemed to be the point of interest.

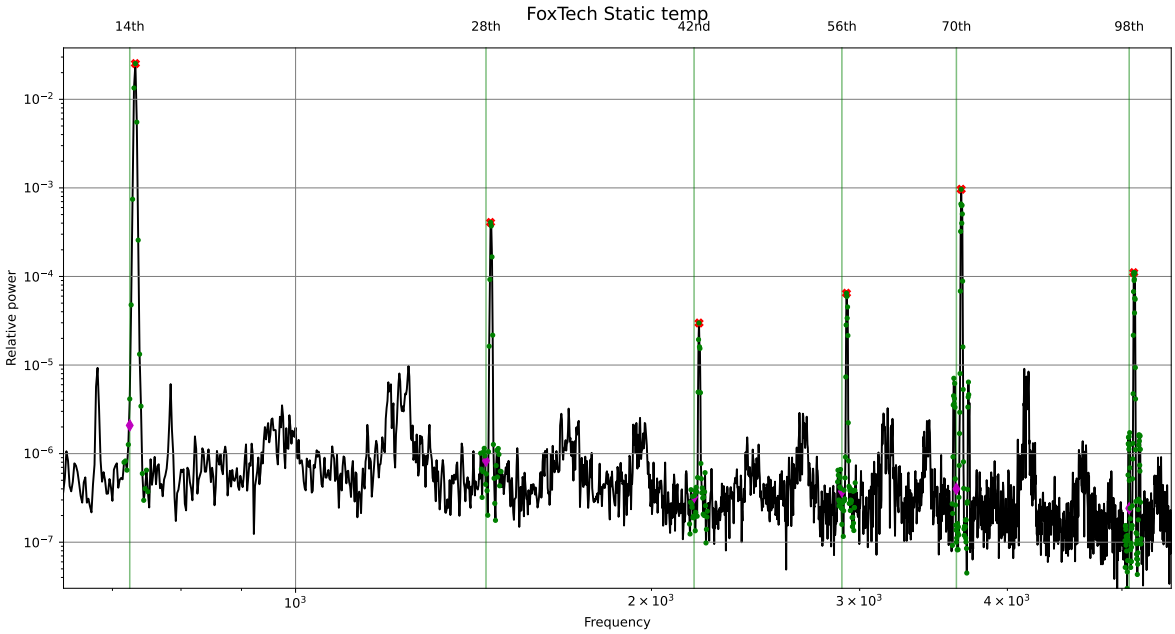


Figure 5.14: Harmonic min/max finding algorithm visualized the electrical harmonics on a normalized three-phase motor current. The Green dots are the intervals k_1 to k_n and the green line is the n_{th} harmonic the purple diamond is the median of the values in k and the red cross is the value of the algorithm has deemed to be the point of interest.

6

Results

This chapter presents the results from tests done using the test rig described in chapter 5 and the methods detailed in chapter 2. First, there will be an introduction to the graphs that will be used to present the test data. Further presenting the method validation tests and moving on to tests designed to find some correlation between the features investigated and the imbalance of the propeller or wear on the motor. In table 6.1 the naming for the damaged and undamaged motor is presented, this will be relevant for the bearing damage tests detailed in section 5.3.5. The motors are the same type of motor, but not the exact same unit, this might affect the results. The reader should be aware that the standard criteria as referenced at the beginning of this chapter and in section 5.3.1, were not strict enough, this will be further discussed in section 6.3.3.

6.1 About the graphs and tables

When reading the graphs note that in the PSD plots, the y-axis is logarithmic and represents relative power, this is an arbitration of the intensity or power a particular frequency has in the time domain signal. The x-axis is representing frequency and also has logarithmic scaling. This means that small distances in either x or y in the plot will mean very large changes to the real value. Also when referring to

Table 6.1: Table of motors and what damage has been done to the motors.

Motor	Damage
1	none
2	lower bearing (ref. 5.10)

harmonics, there are two types discussed, mechanical and electrical. The mechanical harmonics are simply the nth multiple of the mechanical rotational speed, so if the rotor rotates at 50Hz the 3rd harmonic would be at 150Hz. The electrical harmonics are the nth multiple of the electrical rotation speed. The electrical rotation speed is decided by the mechanical speed and the number of pole pairs in the motor, having 14 pole pairs we have for 50Hz rotor speed the 3rd electrical harmonic would be 2100Hz ($50\,\mathrm{Hz} \cdot 14 \cdot 3$). The 1st, 2nd, 3rd, and so forth electrical harmonics are also referred to as the 14th, 28th, and 42nd harmonic as they are the 14th multiple of all the mechanical harmonics. For the comparison tests *ref* denotes the reference test and *test* denotes the test that is being compared to the reference. The PSD is computed using Welch's method with Hann window 50% overlap and 2hz bin size in all the plots if not otherwise specified.

When reading the tables, there are usually a reference value, a test value, and a column comparing the two. These are multiples for the values from PSD analysis with logarithmic spacing, and a difference for the linearly spaced values. This log representation makes it easier to conceptualize the logarithmic values. In the tables, these are usually presented in a test/ref column. For the linear values, there is a difference column.

Lastly in the trend tests the harmonics and time domain features are tracked over a series of tests, this way there is one value for every harmonic and one value for every time domain feature per test. That means for every test there is one point in every subplot for that series of tests.

6.2 Method validation tests

This set of tests is done as detailed in section 5.3.3.

6.2.1 Mad propeller and motor 1

This test was using motor 1 and the Mad propeller. The motor and propeller were first run without any modification as a reference. After this another test was run after a piece of tape measured to be 0.66g and about 1x2cm in size was fastened to the outer part of the top of one of the propeller blades. The taped propeller was simulating a very unbalanced propeller to see if there was a change in the PSD. Testing was done following the standard criteria presented in section 5.3.1.

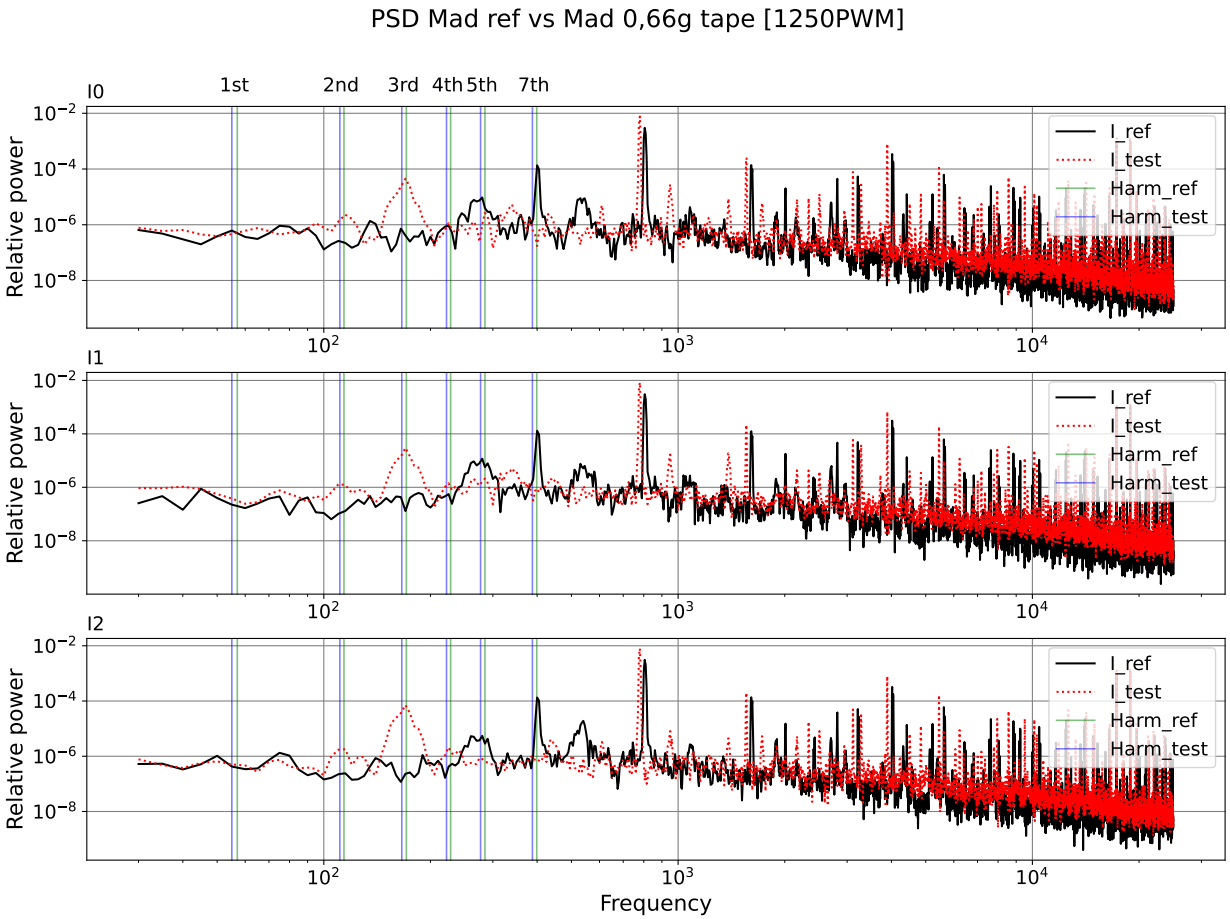


Figure 6.1: Power spectral density plot of the three-phase currents using motor 1 and the Mad propeller with and without 0.66g tape at [1250PWM]. The measured frequency of rotation is for ref 57Hz and for test 55.5Hz.

In figure 6.1, focusing on phase current 1 (I0), one can see that the signal

is quite unchanged in lower frequencies from the reference to the test propeller. Changes do however appear past the 2nd mechanical harmonic, where there is a clear deviance. As discussed in these plots values look smaller than they are, measuring the peaks of the reference and the y value at the same frequency of the test graph they are $3.6e^{-6}$ and $8.3e^{-8}$. This is roughly a 49x increase in relative power.

The next big deviance is in the 3rd mechanical harmonic, the values are $1.06e^{-4}$ and $4.7e^{-8}$. This is an increase of 2255x from the reference to the test value. The 4th and 5th are less extreme. For the 7th the power is interestingly higher in the reference by a factor of about -101117x. Although this test does not show anything in the way of proving a correlation between propeller unbalancedness and the harmonics, there is change in the two frequency spectra. Table 6.2 shows the full change in power for all the mechanical harmonics from both the reference state of the propeller to the unbalanced propeller state.

Table 6.2: Change in power in the mechanical harmonics for Mad propeller vs Mad propeller with 0.66g off-center weight using motor 1.

Mechanical harmonic	I0_ref	I0_test	test/ref
1st	6.287501e-07	1.062413e-07	-5.918130
2nd	8.313517e-08	3.609945e-06	43.422594
3rd	4.718378e-08	1.064108e-04	2255.240849
4th	1.166266e-08	3.470951e-06	297.612341
5th	7.847861e-08	6.912725e-06	88.084193
7th	4.754410e-04	4.701869e-09	-101117.452761

Figure 6.2 and table 6.3 shows the electrical harmonics for the method validation test. The electrical harmonics are definitely more stable between the reference and the unbalanced propeller state. There are however differences in the 2x to 5x range at the most.

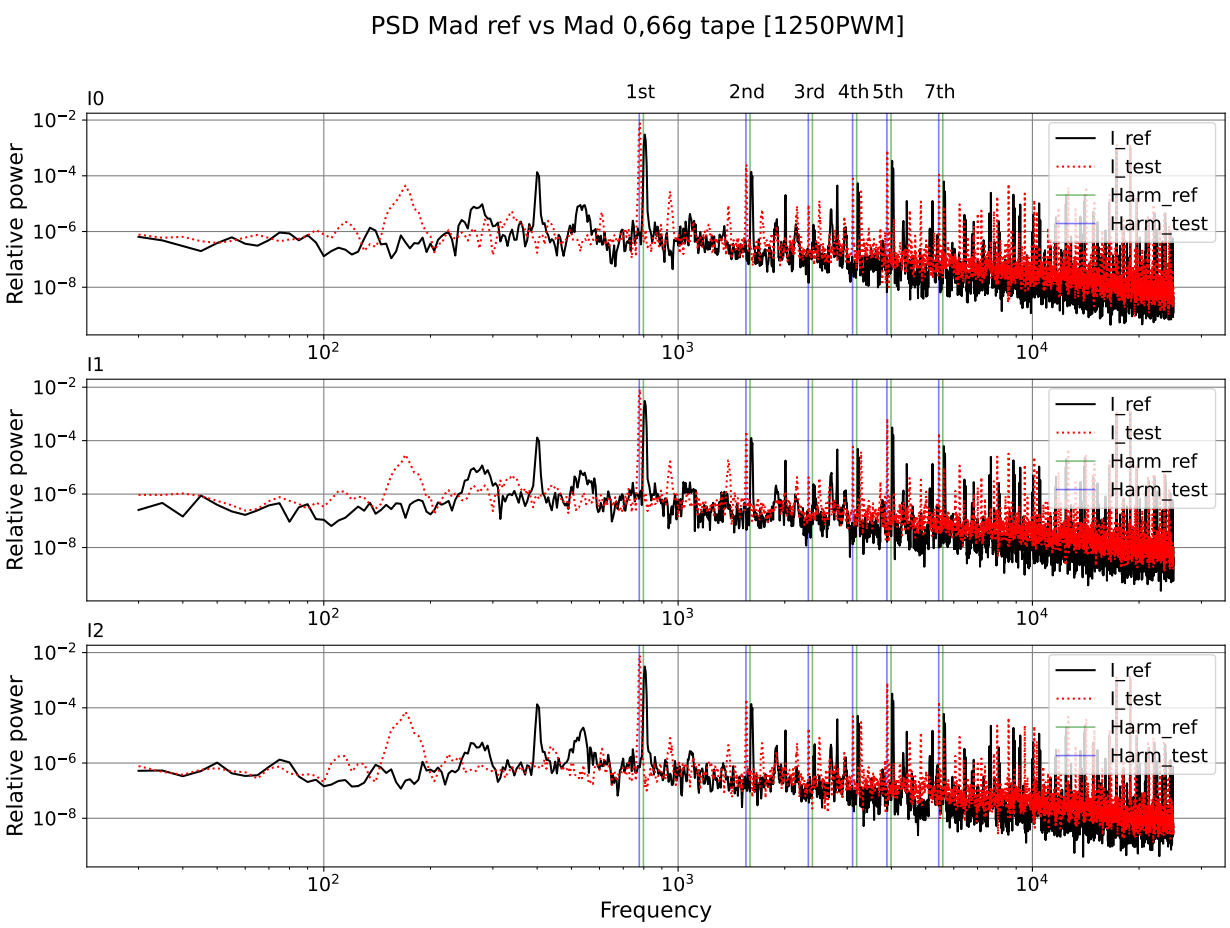


Figure 6.2: Power spectral density plot of the three-phase currents using the Mad propeller and Mad propeller with 0.66g tape at [1250PWM]. The measured frequency of rotation is for ref 57Hz and for test 55.5Hz.

Table 6.3: Change in power in the electrical harmonics for Mad propeller vs Mad propeller with 0.66g off-center weight using motor 1.

Electrical harmonic	I0_ref	I0_test	test/ref
1st	0.008785	0.018021	2.051240
2nd	0.000553	0.000745	1.345642
3rd	0.000004	0.000023	5.986427
4th	0.000141	0.000165	1.169429
5th	0.001105	0.002055	1.860512
7th	0.000197	0.000308	1.567415

6.2.2 Mad prop vs Foxtech

The FoxTech propeller is both bigger and has a more aggressive pitch than the Mad propeller. The effects of this can be seen as a frequency shift to the left in the FoxTech propeller's PSD plot. The higher inertia propeller with more resistance through the air gives more resistance to rotational speed. In turn, this leads to a lower rotational speed at the same PWM frequency as compared to a smaller less pitched propeller. However, interestingly the harmonics are not flat-out higher for one over the other, as shown in figure 6.3 and table 6.4. The main deviance that can be seen from the PSD plot is the lower frequency of the FoxTech propeller. The mechanical harmonics are both up and down on the test subject, this might be because of different damping characteristics of the propellers. The propellers are mechanically quite different as the mad propeller is a folding propeller with an aluminum hub and folding carbon propeller blades while the FoxTech propeller is a stiff all carbon fiber design. An interesting feature in the electrical harmonics is that they are generally higher for the FoxTech propeller. This could be explained by the controller having to put more current through the motor to get a similar RPM.

Table 6.4: Change in power in the mechanical and electrical harmonics for Mad propeller vs FoxTech propeller with no modifications.

Mechanical	Mad	FoxTech	FoxTech/Mad
1st	6.287501e-07	1.227269e-08	-51.231657
2nd	8.313517e-08	7.006842e-09	-11.864855
3rd	4.718378e-08	2.967236e-06	62.886772
4th	1.166266e-08	2.298617e-07	19.709204
5th	7.847861e-08	8.959640e-08	1.141667
7th	4.754410e-04	2.633439e-08	-18054.001029
Electrical			
1st	8.785379e-03	2.264620e-02	2.577715
2nd	5.533344e-04	6.875359e-04	1.242533
3rd	3.855017e-06	7.050314e-05	18.288671
4th	1.413162e-04	1.596361e-04	1.129638
5th	1.104658e-03	1.670110e-03	1.511880
7th	1.966850e-04	1.852127e-04	-1.061941

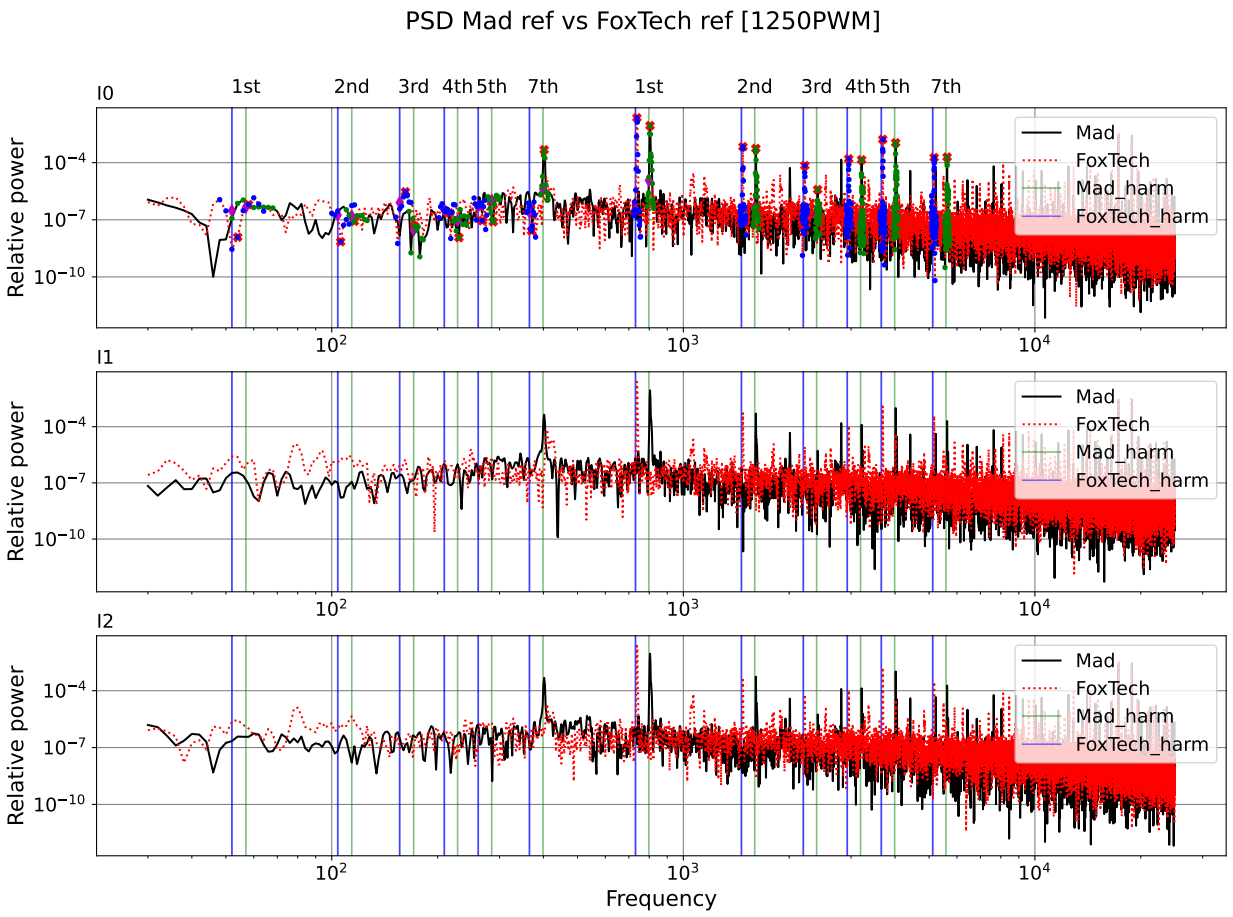


Figure 6.3: Power spectral density plot of the three-phase currents using the Mad propeller and FoxTech propeller at 1250PWM. Mechanical harmonics and electrical harmonics are annotated. The points of the harmonic finding algorithm are also shown in the first phase.

6.2.3 Mad Prop vs Tri-blade

Running this analysis a bug occurred where the PSD would not run using a 2Hz bin with, the test is thus run with a 3Hz bin width instead. In the comparison we can see that the mechanical harmonics are very different from one propeller to another, figure 6.4. This massive change makes sense as the propellers are different both in the number of blades and in the pitch of the blades. Seeing the general increase in power over the spectrum points to a higher current draw at a similar RPM. The changes in the electrical harmonics are however quite small with the only large difference being the 3rd harmonic. The sudden change in the 3rd harmonic can be seen in the graph to be an outlier where the algorithm chooses the wrong point in the tri-blade for comparison. When investigating the graph one can find that the correct point for the 3rd electrical harmonic would be $6.1e^{-}6$. This makes the difference way smaller at only -3.38x instead of the -609x that the algorithm shows.

Table 6.5: Change in power in the mechanical and electrical harmonics of phase 1 for Mad propeller vs Tri-blade with no modifications. The values are sampled from a PSD computed using Welch's method with Hann window 50% overlap and 3hz bin width.

Mechanical	Mad	Tri-blade	Tri-blade/Mad
1st	1.348641e-08	1.112153e-06	82.464729
2nd	4.820712e-10	1.751179e-07	363.261454
3rd	9.102928e-09	7.994848e-06	878.272142
4th	1.115742e-07	1.384893e-07	1.241230
5th	3.338047e-08	1.427703e-05	427.706129
7th	4.971227e-04	2.709052e-07	-1835.043087
Electrical			
1st	7.099057e-03	6.430799e-03	-1.103915
2nd	5.013326e-04	4.885403e-04	-1.026185
3rd	1.865442e-06	3.061840e-09	-609.255045
4th	1.299043e-04	1.226516e-04	-1.059133
5th	8.418660e-04	9.792875e-04	1.163234
7th	1.811164e-04	1.385325e-04	-1.307393

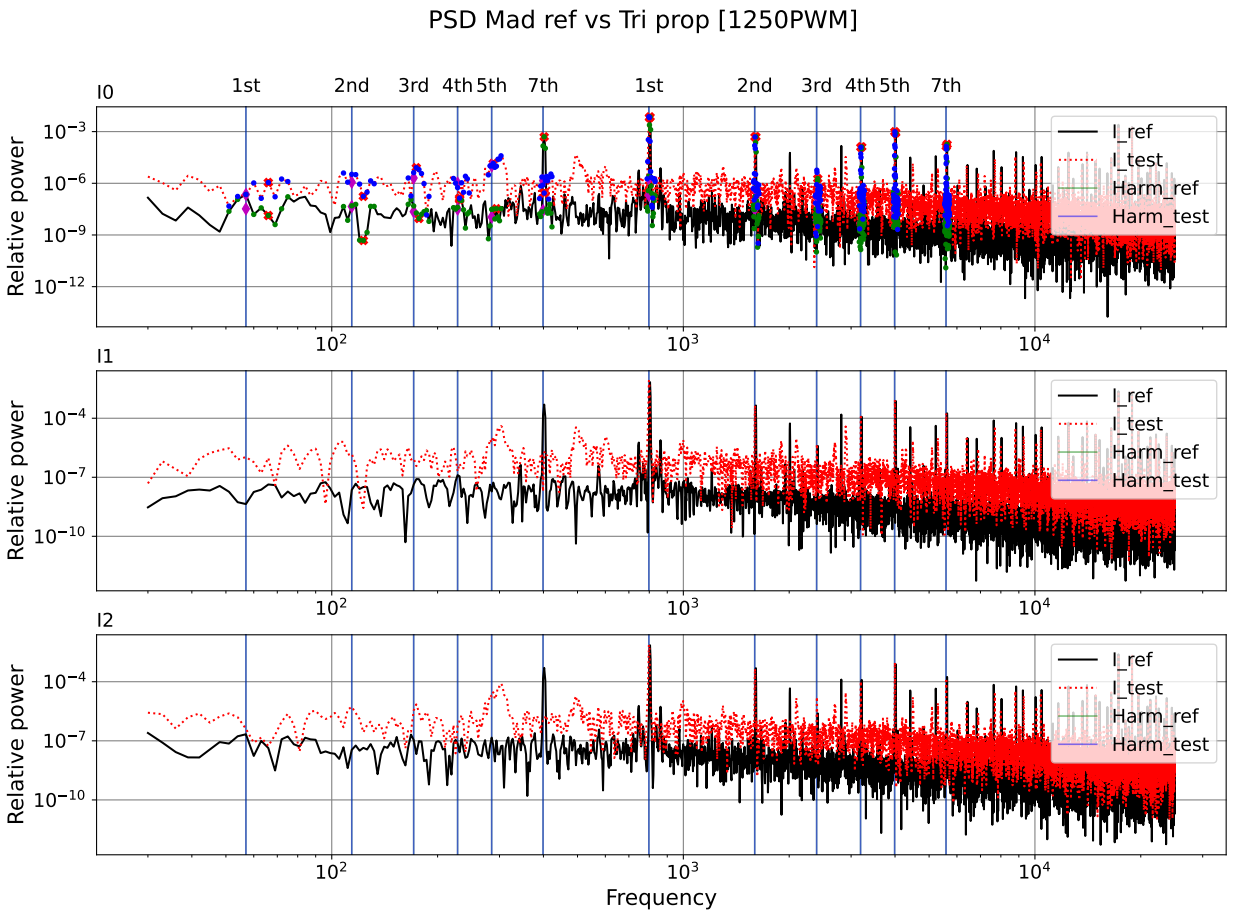


Figure 6.4: Power spectral density plot of the three-phase currents using the Mad propeller and FoxTech propeller at 1250PWM. Mechanical harmonics and electrical harmonics are annotated. The points of the harmonic finding algorithm are also shown in the first phase.

6.3 Trending tests

6.3.1 First trending test

The test was using the Mad propeller and a different amount off weight on the outer 2cm of the propeller. To add weight to the propeller, tape was used. The results show little to no trending, there is an interesting path on the 14th harmonic in figure 6.5 where it shows a small degree of trending. The trend is however ruined at around 0.8g where the power dips below that of 0.0g. This strange effect might be due to the tape not being mounted correctly, the particular test using 0.83g had a loose corner after the test. This might attenuate some and increase other frequency ranges. This flaw in the test seems to show in the 2nd, 3rd, and 7th harmonic as sudden changes in power. The tests in this section are also ranging in temperature.

To rule out the possibility of errors in the algorithm contributing to the strange pattern in for example the 14th harmonic for 0.1g, 0.2g, and 0.3 grams, comparison plots were run and the algorithm was checked to see if it choose the right point to find the true 14th harmonic. It was in fact choosing the right point.

From the time domain features shown in figure 6.6 we also see little to no trending. The RMS seems to trend if the 0.1g, 0.2g, and 0.3g tests are omitted, this could be explained by temperature. To investigate this hypothesis the time domain features are plotted in figure 6.7, but using temperature on the x-axis instead of the weight. This plot also gives little in the way of a solid trend either way, partly because the temperature range is rather small. Also, since there are multiple data points for each temperature, the graph is overlapping itself. The graph makes it clear that varying both temperature and weight makes it hard to determine what parameter changes the features.

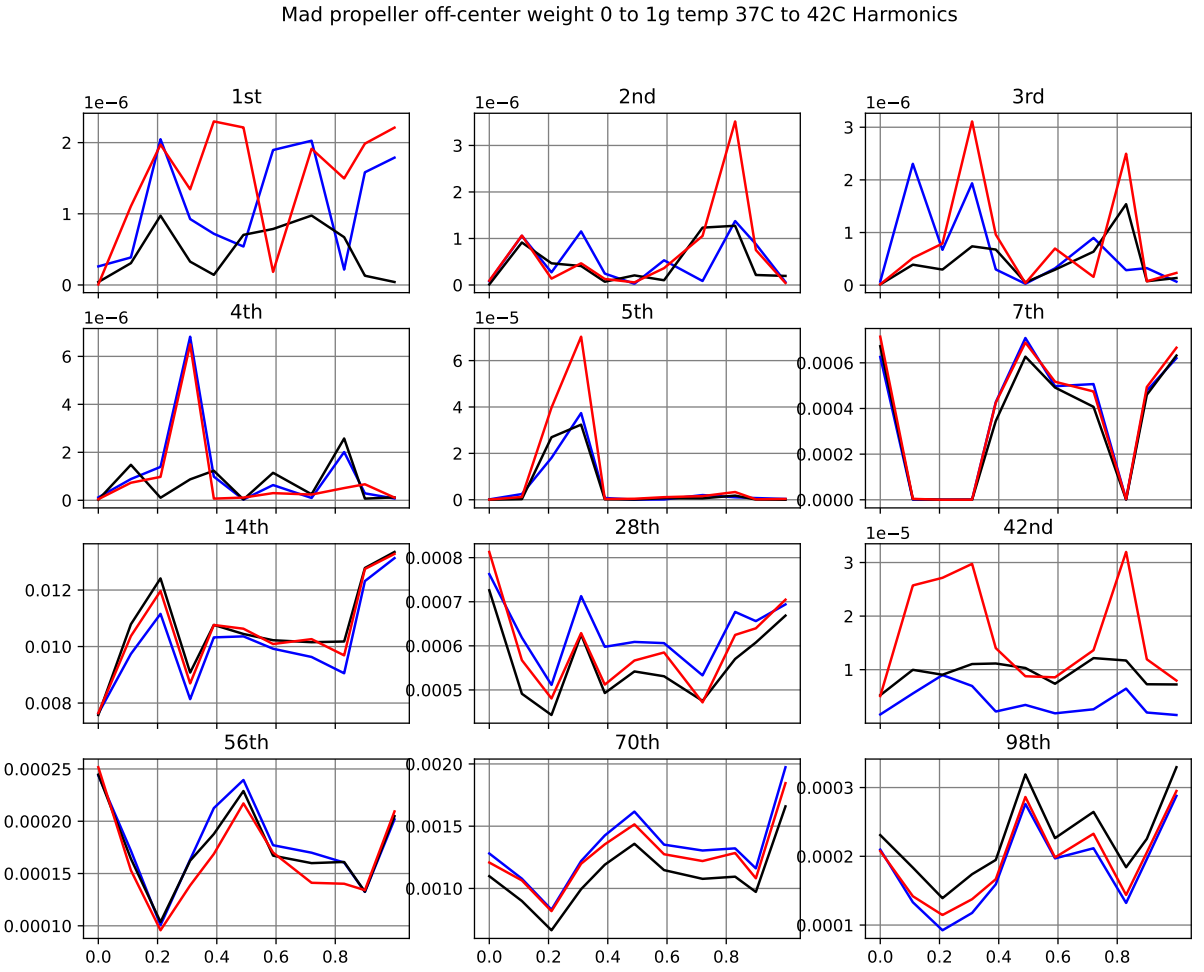


Figure 6.5: Mad propeller with 0g to 1g of off-center weight. Temperature ranging from 37C to 42C, weight is plotted on the x-axis, and y-axis indicates relative power.

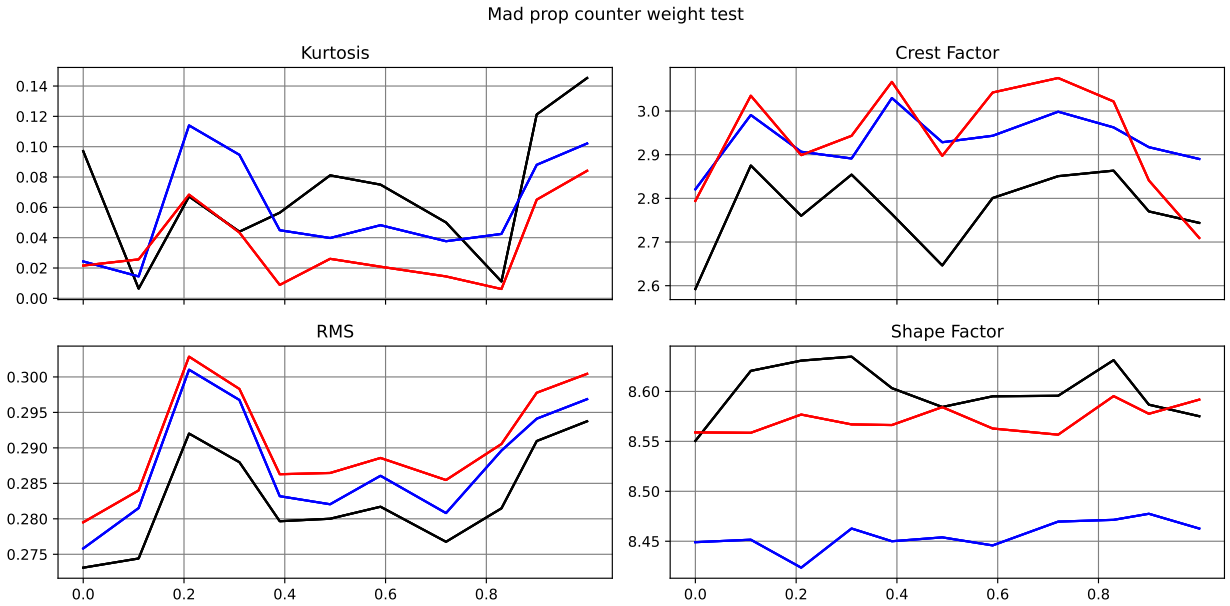


Figure 6.6: Time domain analysis of the three individual phase currents with varying amounts of weight. The weight is ranging from 0 to 1 gram in uneven intervals, plotted on the x-axis.

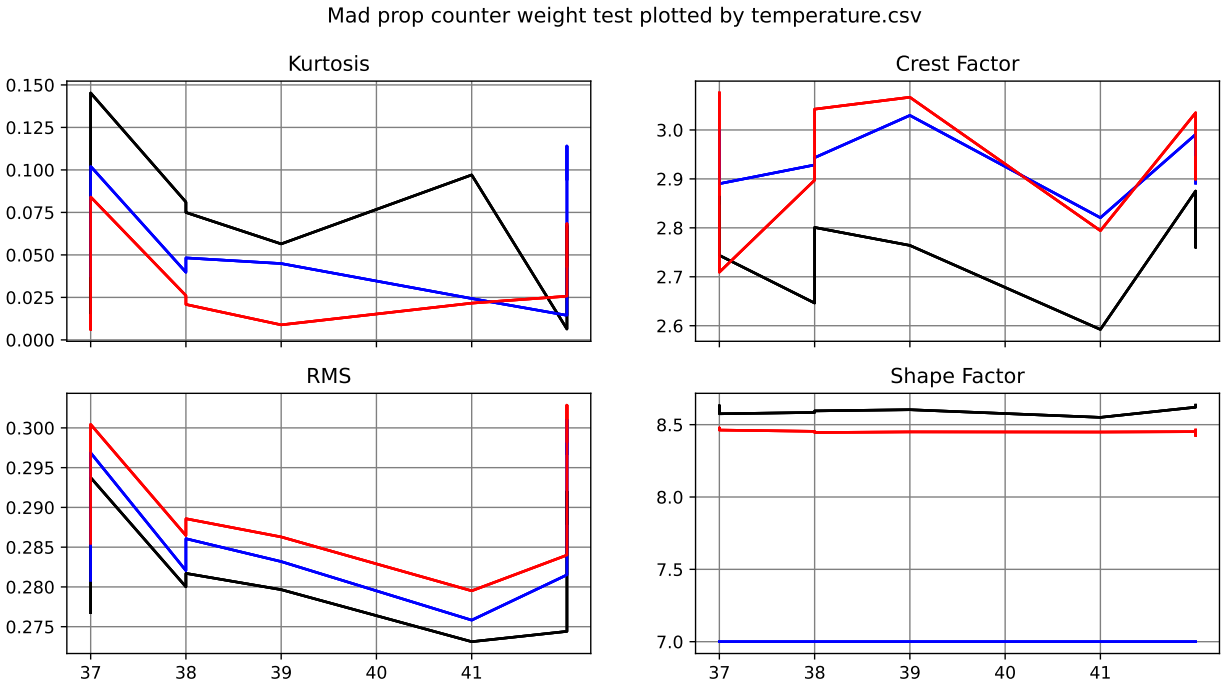


Figure 6.7: Time domain analysis of the three individual phase currents with varying amounts of weight plotted by the measured temperature on the x-axis.

6.3.2 Second trending test

Since the trending was not as expected in the first trending test, the second was based on the hypothesis that the amount of weight on the propeller was simply too much. Having such a light propeller and relatively much weight on the outer part of it could give unknown results.

This test shows no trending at all looking at figure 6.8. The harmonics go up and down at will and no real estimation of how unbalanced the propeller is can be drawn from it.

Further investigation into the time domain features in figure 6.9 show that little to no change can be seen with such a small weight difference. The change from one test to another is negligible, note also that the temperature differences on these series of tests are smaller than the one in section 6.3.1, this will likely also make the differences smaller.

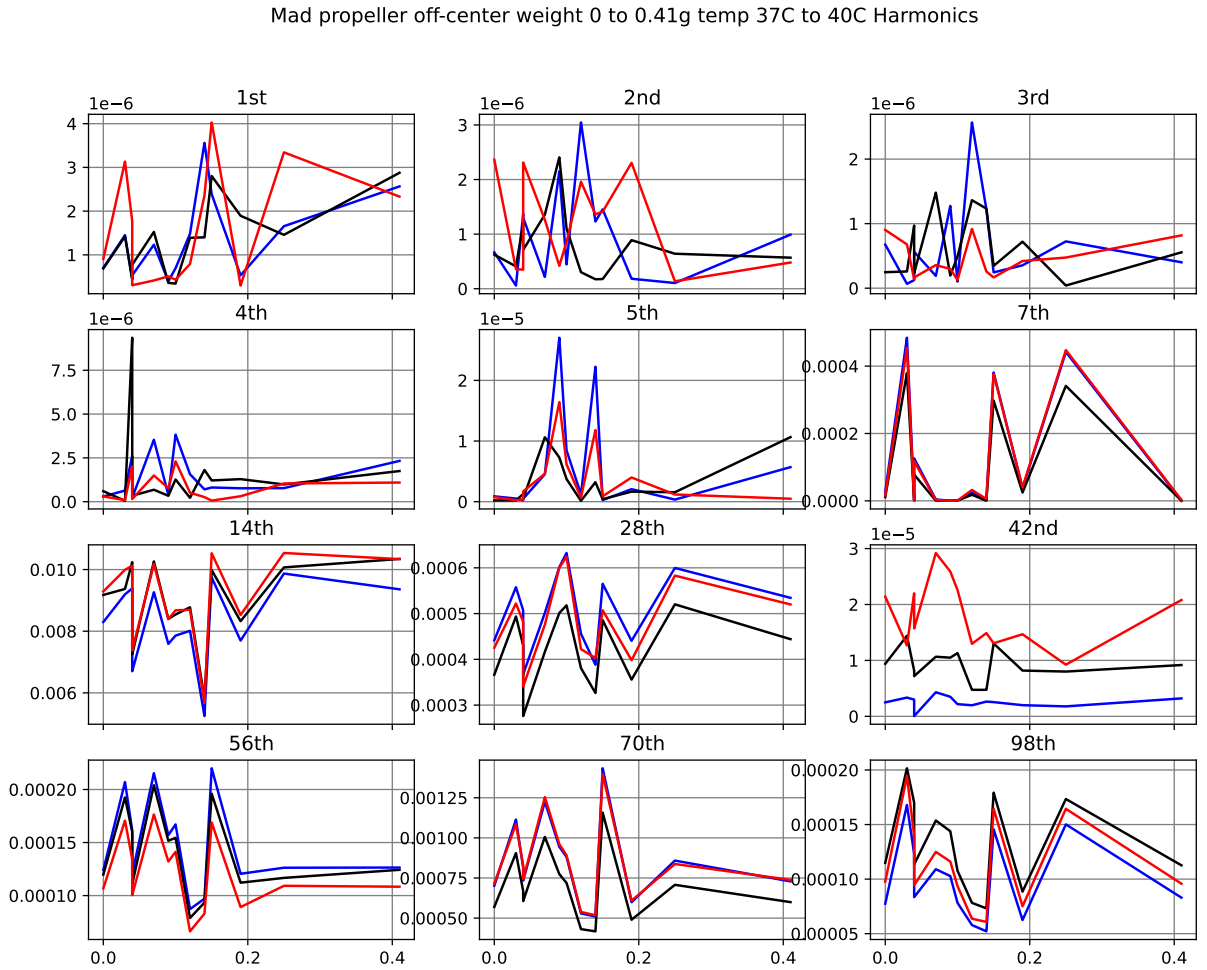


Figure 6.8: Mechanical and electrical harmonics for Mad propeller counter weight test. Weight ranges from 0g to 0.41g on the x-axis, y-axis indicates relative power. All three phases are plotted.

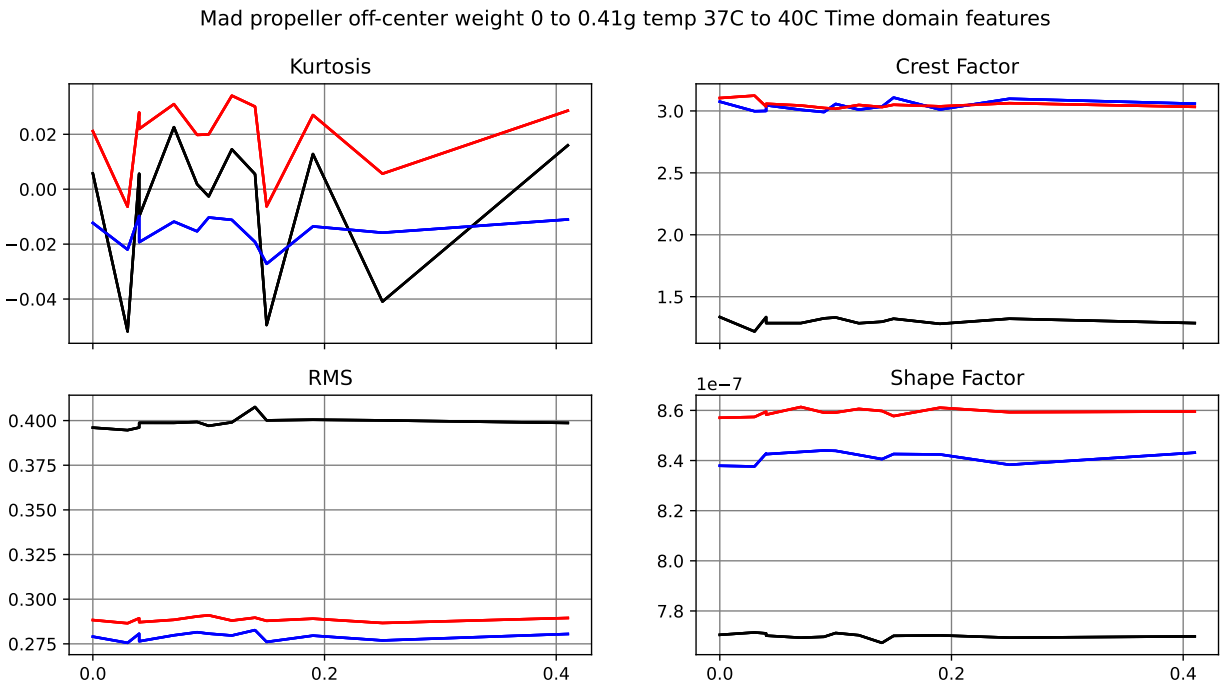


Figure 6.9: Time domain features for Mad propeller counter weight test. Weight ranges from 0 to 0.41g on the x-axis. All three phases are plotted.

6.3.3 Temperature test

The test was done with the FoxTech propeller. The reason for switching propellers was to eliminate the unknown effects of using a folding propeller, which has joints on each propeller blade. The hope was that more consistent results would come from the use of a simple stiff propeller. The temperature test is meant to isolate the temperature as the only changing parameter from test to test.

The test shows a better trending than any other test so far in the time domain features. This result indicates that the changes in temperature from one test to another potentially has more effect on the features that have been picked out than off-center weight does. Shape factor is not showing any signs of change, but kurtosis, crest factor, and RMS seem to have a close to linear relationship with an change in temperature.

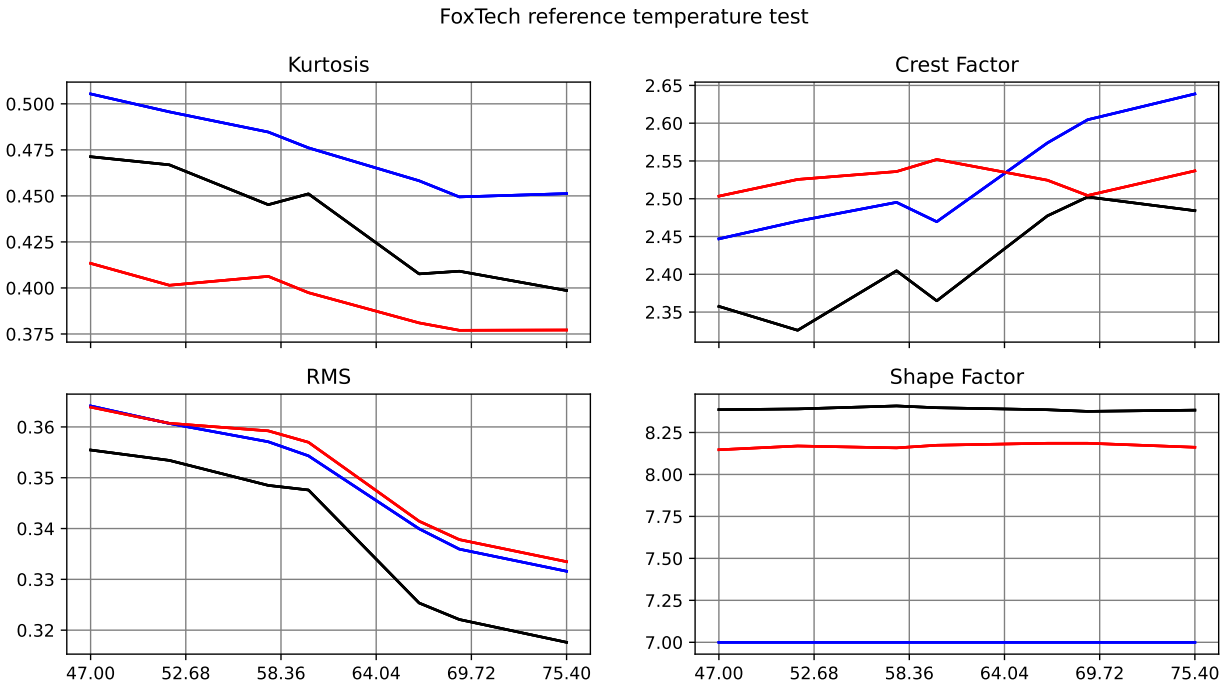


Figure 6.10: Time domain features over temperature from 75.4C to 47.0C on the x-axis. Showing all three phases in all plots.

The harmonics during the test show few interesting features. In the electrical

harmonics, the power decreases sharply after about 67C. One explanation could be the change in viscosity of the bearing grease, this effect would however likely be more linear than this step.

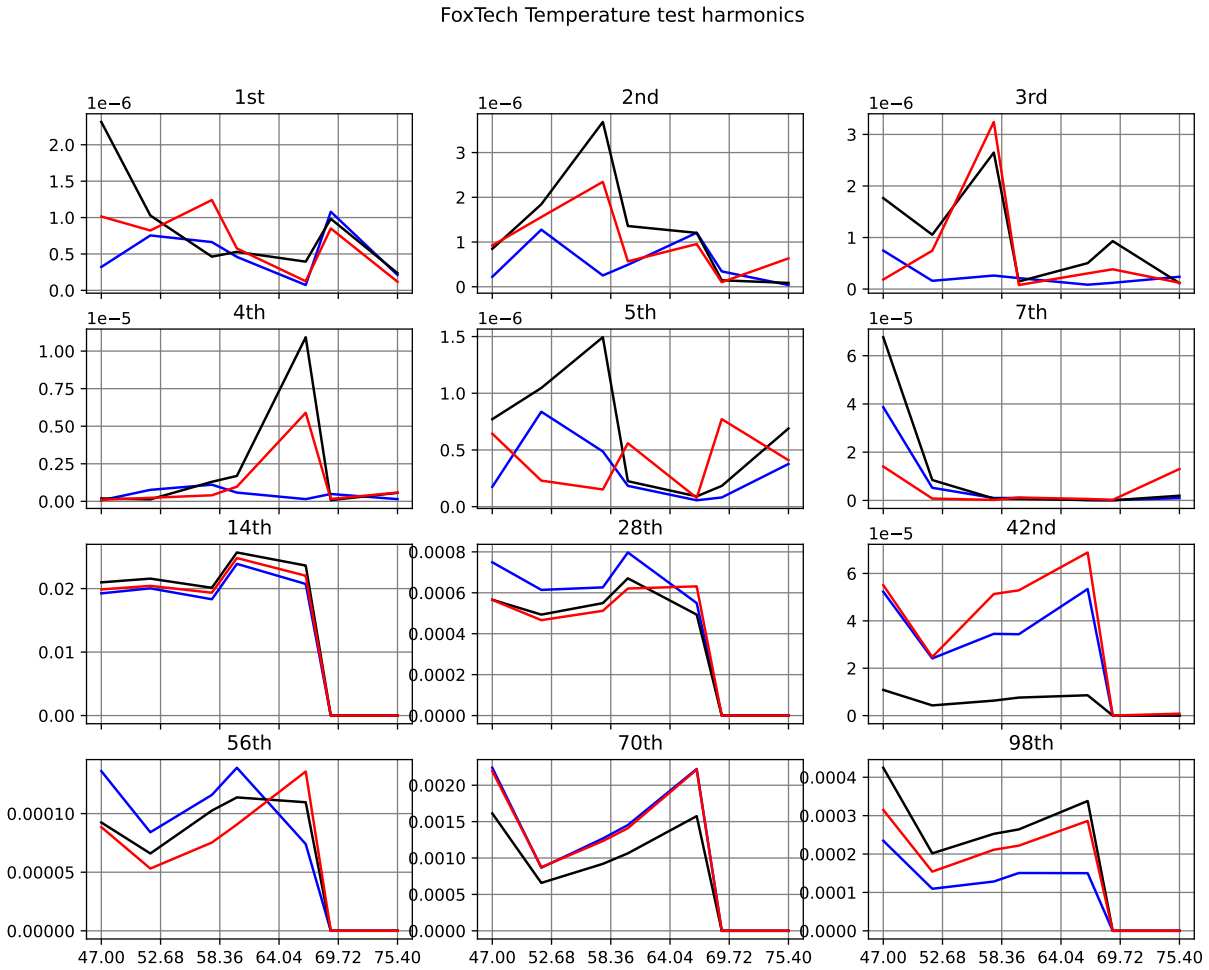


Figure 6.11: Frequency domain features over temperature from 75.4C to 47.0C on the x-axis. Showing all three phases in all plots.

6.3.4 Third trending test (constant temperature)

First showing the most interesting, figure 6.12 shows the first real trend with weight as the test parameter. All of the time domain features, except crest factor, show trending with an increased weight either going up or down in a rather linear fashion. One of the more linear is the shape factor. This gives promise to it being a good feature to track as this feature is the only one changing very little in the temperature test in section 6.10.

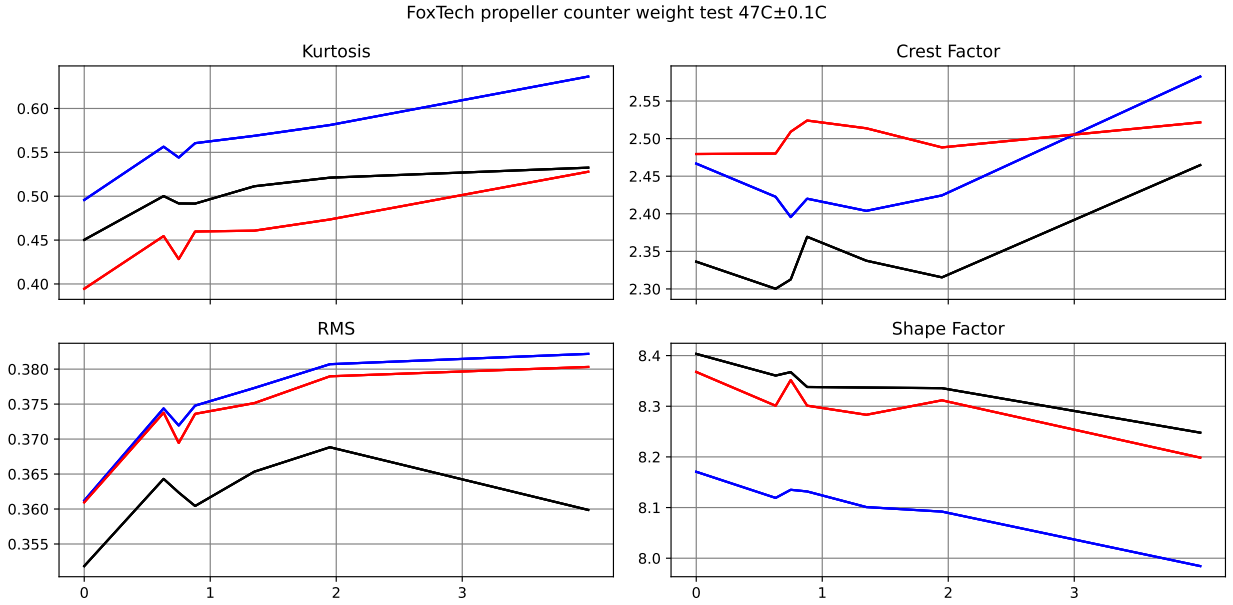


Figure 6.12: Time domain analysis of the three individual phase currents with constant temperature on the motor (Measured at lower bearing 47C) using FoxTech propeller and a varying amount of off-center weight on the propeller. The weight is ranging from 0 to 4 grams in uneven intervals on the x-axis.

The same analysis was run on a normed sample of the phase currents of this test as an attempt to reduce complexity in computation and data handling, the result is shown in figure 6.13. The result shows a similar story to the three-phase plot. As expected the phase currents that are trending more are somewhat affected by the worst trending phase. This effect does however seem to be less of a problem since there seem to be two phases trending well and one a bit worse for for the features in this particular test.

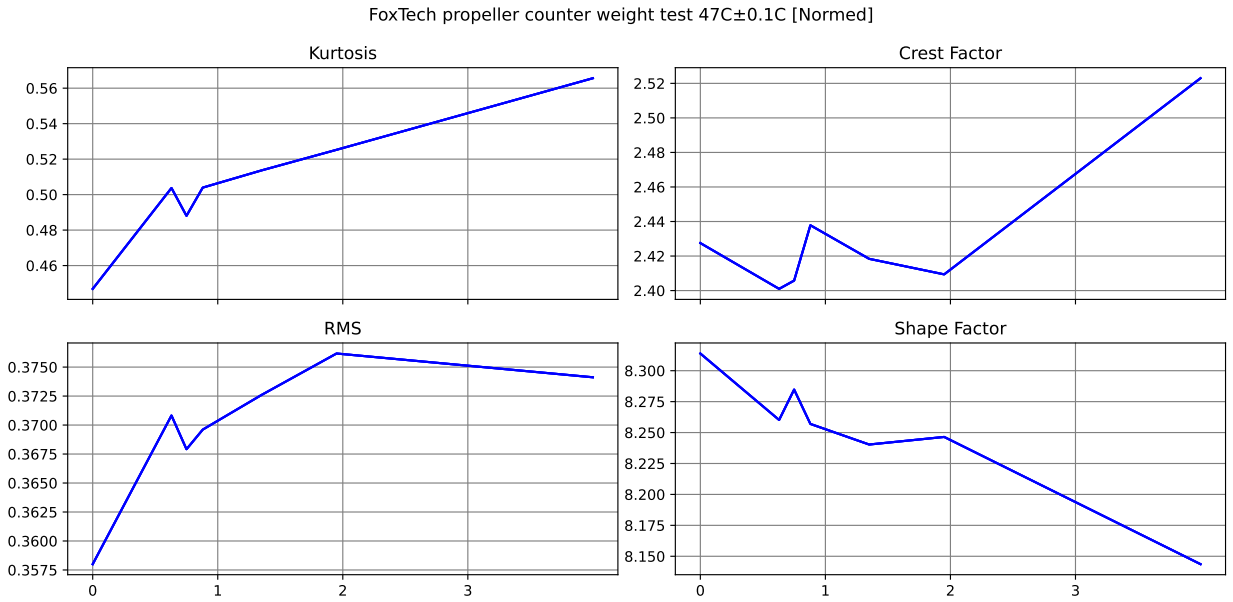


Figure 6.13: Time domain analysis of the three normed phase currents with constant temperature on the motor (Measured at lower bearing 47C) using FoxTech propeller and a varying amount of off-center weight on the propeller. The weight is ranging from 0 to 4 grams in uneven intervals on the x-axis.

While the time domain features were showing promise the harmonics are not as consistent. Examples of this inconsistency can be seen in the 14th harmonic (2nd electric) where the 0.7g and 0.9g have as high value as the 4g test.

Removing all tests that were off by 0.1C or more, the series has more stable values in the graph shown in figure 6.15. It does however leave very few points making this plot less relevant.

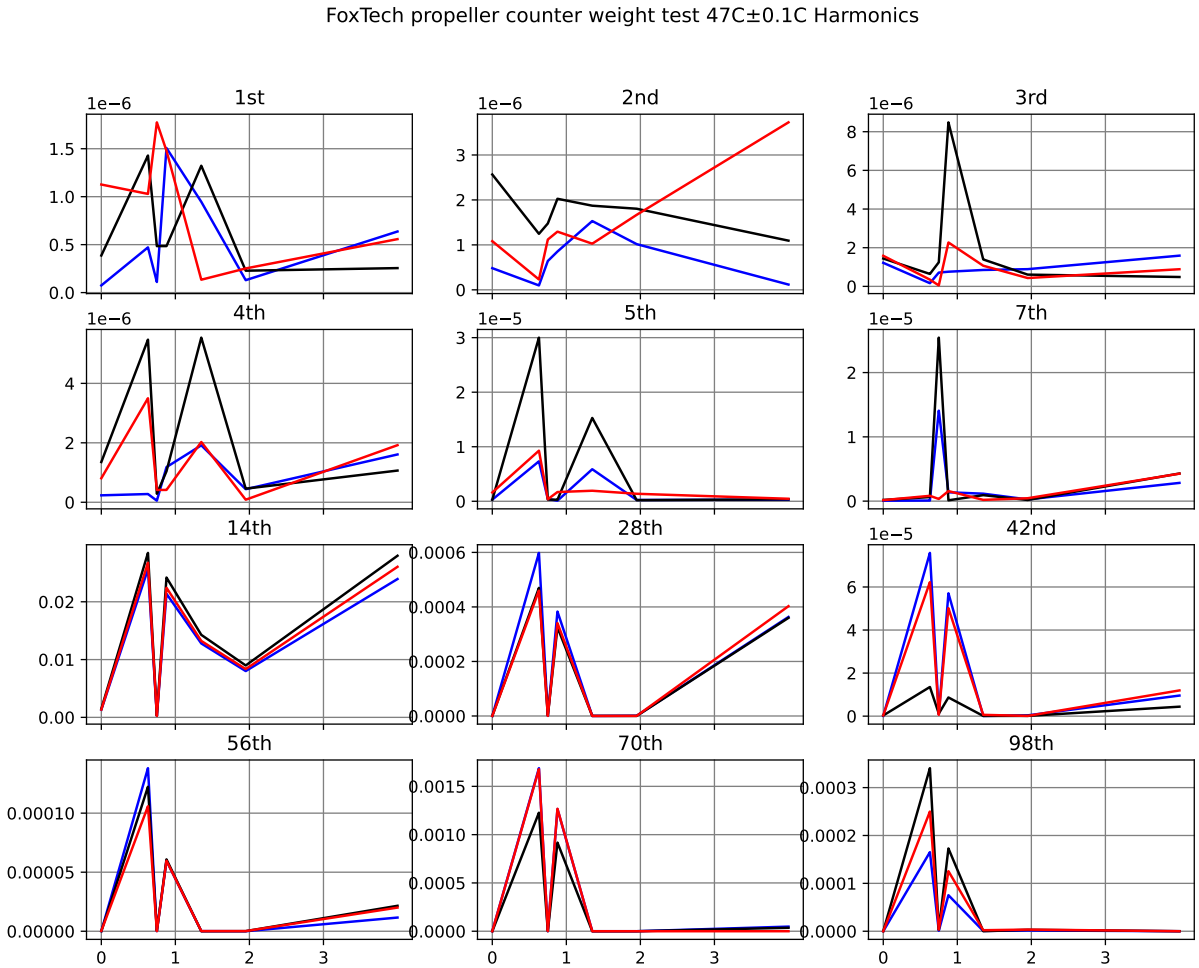


Figure 6.14: Harmonics of test with constant temperature and FoxTech propeller.

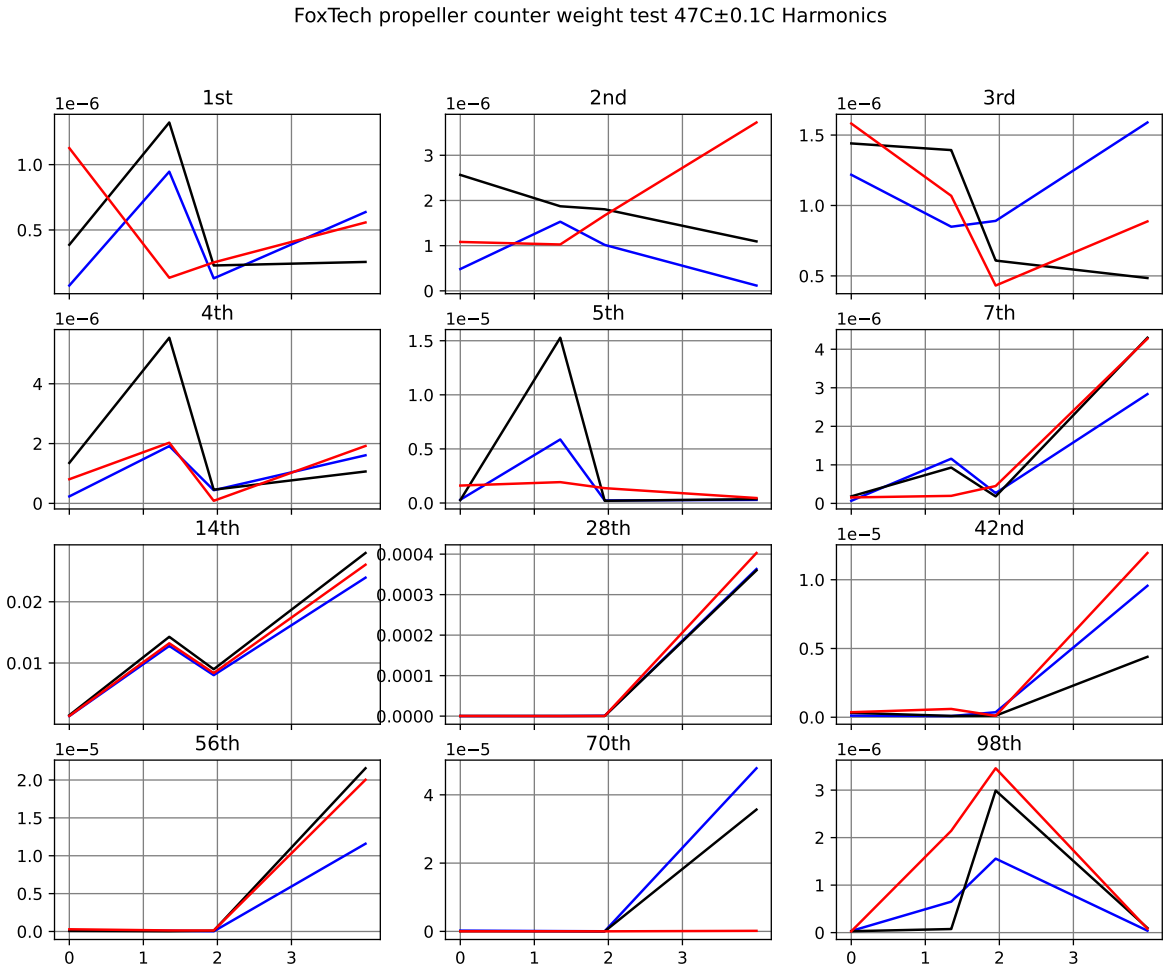


Figure 6.15: Harmonics of test with constant temperature and FoxTech propeller. Removing the tests where the temperature was 0.1 or more off the target of 47 reveals some correlation.

6.4 Bearing damage test

This section presents test results from before and after the lower bearing in the motor was damaged as described in section 5.3.5. The data is gathered from both the Mad and the FoxTech propeller. This data was gathered before finding the large influence of temperature on the motor current.

6.4.1 Mad propeller

Investigating the graphs in figure 6.16 we see a large deviation by the 3rd harmonic both in the mechanical and the electrical. It does also appear that this change is more pronounced in the first and third motor current phase. The 7th mechanical also sees a large change, this change can be seen in all the phases presented. The increase in the 4th mechanical harmonic seems to be affected by the general increase in power around the 3rd, it is also obvious that the algorithm is picking a sub-optimal point as the harmonic, this is shown in detail in appendix A. The other harmonics seem correctly picked out.

When calculating the different common frequencies of defects presented in section 3.3.5, we get:

$$f_{BD} \approx 95.375\,\mathrm{Hz}$$
$$f_{OD} \approx 47.6875\,\mathrm{Hz}$$
$$f_{ID} \approx 145.125\,\mathrm{Hz}$$
$$f_{CD} \approx 47.6875\,\mathrm{Hz}$$

Using the values calculated for f_{BD}, f_{OD}, f_{ID}, and f_{CD} we find a feature that coincides with these values. The f_{ID} coincides with a sharp downward spike a little to the left of the 3rd harmonic. In table 6.6 there are large changes from motor 1 to motor 2 all but the 5th mechanical harmonic. The electrical harmonics are mainly differentiated by the 5th.

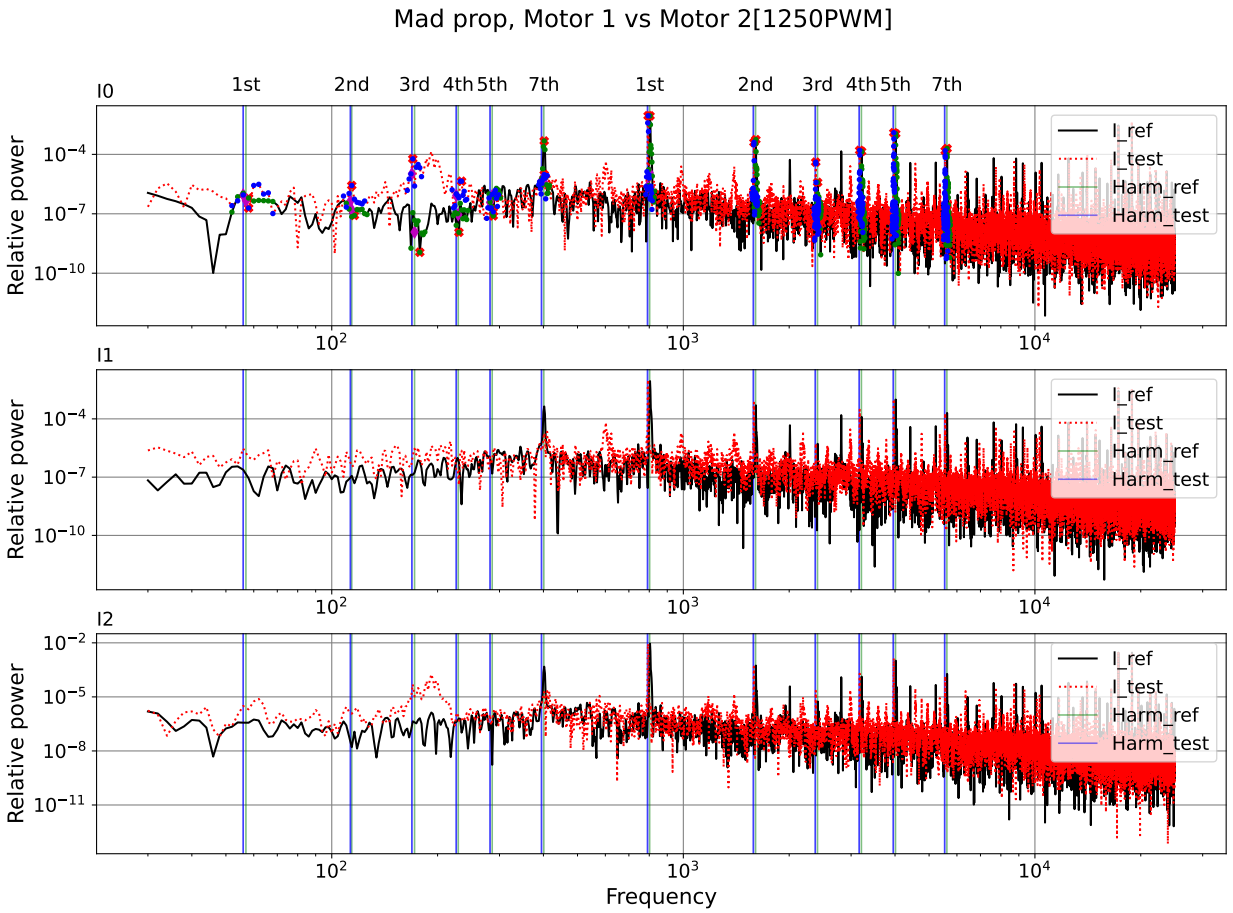


Figure 6.16: Bearing test using Mad propeller plotting result from motor 1 and motor 2. Harmonics are shown for both the reference measurement and the test measurement.

Table 6.6: Change in power in the mechanical and electrical harmonics of phase 1 for motor 1 vs motor 2 using the Mad propeller.

Mechanical	M1	M2	M2/M1
1st	6.287501e-07	1.957850e-07	-3.211432
2nd	8.313517e-08	2.679058e-06	32.225330
3rd	1.141615e-09	5.959850e-05	52205.419496
4th	1.166266e-08	4.412556e-06	378.349105
5th	7.847861e-08	1.255964e-07	1.600390
7th	4.754410e-04	9.539952e-06	-49.836839
Electrical			
1st	8.785379e-03	8.901497e-03	1.013217
2nd	5.533344e-04	3.997884e-04	-1.384068
3rd	3.855017e-06	4.048093e-05	10.500844
4th	1.413162e-04	1.594101e-04	1.128039
5th	1.104658e-03	1.241409e-03	1.123796
7th	1.966850e-04	1.535659e-04	-1.280786

Table 6.7: Time domain features for the individual phases in the damaged bearing test. Using the Mad propeller for the test with motor 1 and motor 2.

Feature_phase	Motor 1	Motor 2	Difference
Kurtosis_0	-0.019	0.114	0.133
Kurtosis_1	-0.010	0.020	0.030
Kurtosis_2	0.028	-0.039	-0.067
Crest Factor_0	2.920	3.174	0.254
Crest Factor_1	3.097	3.085	-0.012
Crest Factor_2	3.113	2.840	-0.273
RMS_0	0.276	0.269	-0.007
RMS_1	0.283	0.284	0.001
RMS_2	0.282	0.264	-0.018
Shape Factor_0	1.2712e-06	8.051e-07	-4.661e-07
Shape Factor_1	1.2343e-06	7.965e-07	-4.378e-07
Shape Factor_2	1.2598e-06	7.967e-07	-4.630e-07

6.4.2 FoxTech propeller

Testing motor 1 against motor 2 using the FoxTech propeller there are about the same magnitude of changes in the electrical harmonics. The mechanical harmonics have a drastic flattening of the 7th harmonic shown in table 6.8 and figure 6.17. The calculated frequencies for f_{BD}, f_{OD}, f_{ID}, and f_{CD}, the frequencies have no large coinciding features. The time domain features in table 6.9 show a similar story as the Mad propeller time domain features did. The largest changes are in Crest factor and Kurtosis.

$$f_{BD} \approx 88.472\,\text{Hz}$$
$$f_{OD} \approx 44.236\,\text{Hz}$$
$$f_{ID} \approx 134.720\,\text{Hz}$$
$$f_{CD} \approx 44.236\,\text{Hz}$$

Table 6.8: Change in power in the mechanical and electrical harmonics of phase 1 for motor 1 vs motor 2 using the FoxTech propeller.

Mechanical	M1	M2	M2/M1
1st	3.212237e-07	2.824997e-07	-1.137076
2nd	2.223128e-07	1.344732e-07	-1.653213
3rd	7.468119e-07	3.052725e-07	-2.446378
4th	6.992264e-08	1.305485e-06	18.670416
5th	1.750561e-07	1.894022e-06	10.819512
7th	3.866788e-05	2.298536e-07	-168.228275
Electrical	**M1**	**M2**	**M2/M1**
1st	1.925008e-02	2.595378e-02	1.348243
2nd	7.491589e-04	6.121057e-04	-1.223905
3rd	5.227291e-05	6.617312e-06	-7.899417
4th	1.362288e-04	1.157085e-04	-1.177344
5th	2.240180e-03	1.772536e-03	-1.263827
7th	2.347249e-04	2.406301e-04	1.025158

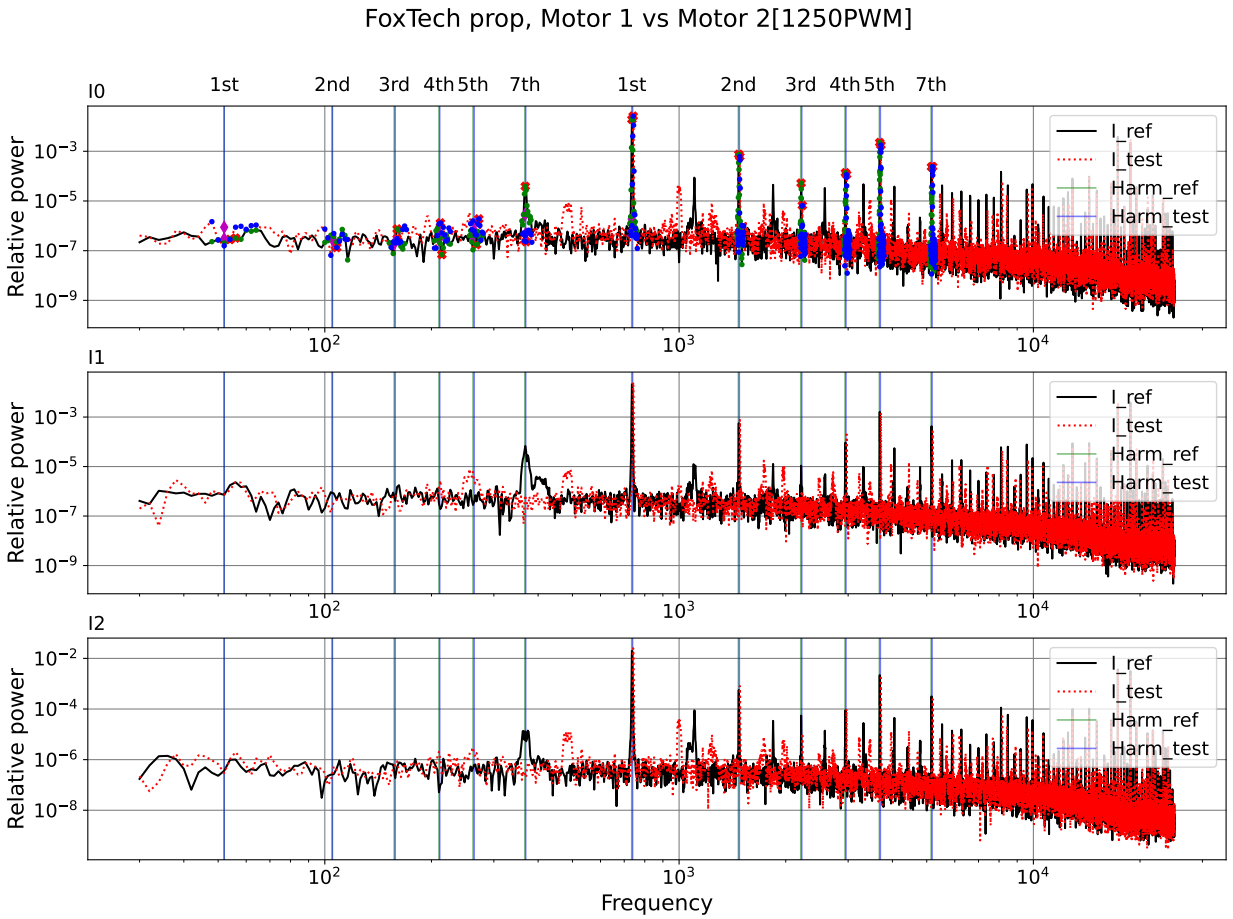


Figure 6.17: Bearing test using FoxTech propeller plotting result from motor 1 and motor 2. Harmonics are shown for both the reference measurement and the test measurement.

Table 6.9: Time domain features for the individual phases in the damaged bearing test. Using the FoxTech propeller for the test with motor 1 and motor 2.

Feature_phase	Motor 1	Motor 2	Difference
Kurtosis_0	-0.471	-0.434	0.037
Kurtosis_1	-0.505	-0.404	0.101
Kurtosis_2	-0.413	-0.523	-0.110
Crest Factor_0	2.358	2.419	0.061
Crest Factor_1	2.447	2.547	0.100
Crest Factor_2	2.503	2.389	-0.114
RMS_0	0.355	0.360	0.005
RMS_1	0.364	0.362	-0.002
RMS_2	0.364	0.354	-0.010
Shape Factor_0	8.385e-07	7.804e-07	-5.81e-08
Shape Factor_1	8.147e-07	7.829e-07	-3.18e-08
Shape Factor_2	8.357e-07	7.723e-07	-6.34e-08

7

Discussion

This chapter will discuss some of the decisions made in the thesis as a whole, it will defend some and criticize others. It will also cover discuss some other effects of the implementation of a CBM strategy in the drone industry.

7.1 Component decision

7.1.1 Current sensor

The decision on sensors was largely based on previous experience and availability. This meant that sensors that were already in-house at Aviant were most of the basis for development. Some sensors, like the bidirectional current sensor ACS770 had to be modified. It was mounted to a breakout board meant for a unidirectional ASC758 in order to measure the negative current and voltage of the system as no breakout boards for this sensor were available at the time of development. The breakout was from a high-voltage current sensor from Mauch Electronic and the job was done based on some instructions from the developer of the board. The solution would be more elegant with a more embedded solution like an ESC with current measurements. Having purpose-made breakouts with the ACS770 would also remove some uncertainty in this component, or with the correct equipment,

the custom solution could be checked and calibrated.

7.1.2 Data logger

When the data logger, the MCC 118 – 12-Bit Voltage Measurement HAT, was not in stock at the time of planning hardware, multiple solutions were tried, this is explained in more detail in sec 7.4. After no good components for an in-drone solution was found, a more static lab setup was deemed best. The solution had to be quick and easy to implement, this led to the use of a digital oscilloscope, the Logic 8 pro. This unit was kindly lent to me by Krzysztof Cisek in Scout Drone Inspection, so it was available on short notice and ultimately meant testing could be done.

Although functioning, this solution was not optimal. Having a data acquisition unit directly connected to the Pi would make the workflow better, and would make it possible to run failure tests autonomously. This was not possible with the Logic Pro 8. If this work is ever reproduced this seems to be one of the best improvements to workflow and test stand possibilities. It would also mean quick integration with the raspberry pi already on the drone for testing.

7.1.3 Other sensors

The other components like the motor, ESC, and raspberry pi that are presented and used in the thesis were already part of the drone design. Such that using them would make a prototype easier to implement on a drone if the progression allowed it.

7.2 Limitations

7.2.1 Test stand

The test stand is not a perfect analog for the drone. Not only is it, not a complete body, but many of the components are also missing, which could affect the tests. It is also put in an inside environment, this can have different impacts. The propeller is pushing air around in a small room which could disturb the airflow around the

propeller and affect the measurements. In the other case, outside there might be wind, and as seen temperature could be an issue.

7.2.2 Sensor placement

Sensor placement does not need to be an issue, but for some of the sensors, especially the units running I^2C this was a huge issue during the work on this thesis. Having the sensors placed by the motor and phase current cables meant the communication between it and the raspberry pi would break down at medium to high RPM. This is suspected to be because of the large amount of electrical noise generated by the high rapid switching currents. Having robust data protocols there are many good and available places to mount sensors mechanically, as can be seen in the designed sensor mounts.

7.2.3 Raspberry Pi and python

There are limitations to using Python and raspberry pi in this thesis. The reason this combination was chosen is that it is already used day to day in Notus. Python in itself is a less efficient programming language than for example C or C++, and the raspberry pi lacks analog input pins. If the onboard companion computer had the same feature set as the raspberry pi with the addition of well-integrated analog to digital converters, the implementation would be way more elegant. The effort of the thesis could then be put into code quality and test development, gaining better insight and better data. This is time better spent than modifying other suboptimal hardware to do the same job.

7.2.4 Motor and ESC combination

Using motors and ESCs with lacking or outright no data sheet available to the public at all, as is the case for the specific MN505-s 260kv, is suboptimal at best. These motors and ESCs are good for their price point, but they are a long way away from a proper high-quality unit like for instance an Alva unit. This makes for uncertainty in the consistency from motor to motor, and also the consistency from one winding to another on the same motor. Having a high-quality starting point is more expensive but seems like a better option for further investigation into

this field in general.

7.3 Method

7.3.1 Problems with temperature and feature analysis

The temperature was always deemed a possible variable, but it was not considered that it could make as large of an impact as it did. Figure 6.10 shows the impact of temperature from testing with a FoxTech propeller using normed phase currents. There could be many reasons for this, both electrical and mechanical. First off the bearings have grease inside, this grease should have different characteristics for different temperatures. The contribution of this change is hard to quantify but could be done in a bearing test rig. This way one could measure the bearing at different temperatures without being influenced by the temperature of the motor. On the electrical side, a motor consists of long stretches of copper wire wound around an iron core. Both copper and iron change electrical characteristics at different temperatures. Since the copper wires in the motor are long, a small change in resistance per cm of the wire could make out a large portion of the change in the signal.

The problem regarding massive changes in the motor current signal at different temperatures could be an inherent flaw of the measurement method. It could be compensated for if a good relationship between the signal and the temperature was found. This problem might be fixed by using accelerometer measurements. However, this could affect especially the ability to measure flaws in the bearings as propeller vibrations might overpower the small bearing vibrations. On this note, it should be taken into consideration that the noise of a bad bearing is often high-pitched and loud. During testing with a damaged bearing, the noise of the damaged bearing could be heard over the sound of the propeller at 1250PWM. This could mean the vibrations are powerful enough, or maybe acoustic measurements could be used.

7.3.2 The algorithm

Although the algorithm developed for this thesis to find the power of the harmonics works for most of the data in the thesis, it isn't perfect. The principle seems to be solid at finding peaks in log-spaced data, and with some adjustment and some more time spent, it could work for more than this specific scenario. For this thesis, the algorithm was not the end product and was therefore deemed good enough for now.

7.3.3 Relying on bad RPM measurements

Relying on bad RPM measurements when finding harmonics and also when calculating the fault frequencies of the bearings will make the end result less precise. Getting good RPM measurements with a higher precision should be a high priority when using these methods.

7.4 Failed attempts and lessons learned

In the search for alternative components to make a solution that was quick to implement and could be fitted to a drone for test flights, there were some promising ideas. These ideas did however end up being either too slow or too complex to implement in such a short time frame. This section covers some of these implementation attempts.

7.4.1 Reading from different analog to digital converters

Simply reading from an analog to digital converter using a common data protocol on the raspberry pi would make for a simple system design. This was tried with multiple modules from different producers. Most of them were running simple I^2C interfaces and most of them were plagued by slow logging, at a maximum of about 9000 samples per second, and communication breakdown when the motor was spinning. The issue was often not with the actual analog-to-digital converter on the board, but with the carrier board. Often the board had lower-speed communication than needed and badly written libraries to go with them. Usually using Python, the libraries for reading the analog values would not be able to open a data stream but

rather initialized and closed the communication from the converting register. This gave a very poor performance.

7.4.2 Reading directly from the registers of an Arduino

Without explaining too much in the way of registers and embedded coding, the Arduino was pushed to around 70k samples per second at 8-bit resolution. This was done by sampling the value in the analog-to-digital converter register before the full 10-bit conversion time. Overclocking the Arduino could have been done to increase the sample rate further, but only by about double or triple, this was still not enough to measure at the speed needed for the application.

7.4.3 STM32 Feather data logger

The STM32 is a very fast small and suitable microcontroller for fast analog sampling. The feather board logger incorporates this microcontroller in a small form factor with a micro SD card reader and some data pins. As it has support for the Arduino IDE and the previous attempt was based on using it, this was deemed the fastest path of progression. It does however seem like the the board is quite new and had little software support in the Arduino IDE. After much tinkering, there was also discovered a hardware issue on the circuit board of the logger as a trace was designed wrong in this specific production batch of the loggers. This meant that the micro SD card detection did not work on a hardware level. The fix was to cut a trace on the board with a scalpel, this fixed the issue of the reader. The support of the Arduino IDE was however still lacking. At the time of trying this approach, no libraries with proper hardware register access were found. Writing embedded code for this was not an option at the time since that could end up being a very time-consuming endeavor.

7.5 Deviation from planned tasks

7.5.1 Run-to-failure

The thesis was originally planned to have run-to-failure tests using the MN505-s, but because of time constraints, tight budgets, and low availability in some key

parts, this was not done. Too little time was left when the test stand was operational to a point where there was trust in the data it put out. This did however lead to other findings like temperature having a large influence on the time domain features and that the effects to a large degree can be avoided by strict temperature control.

7.5.2 Migrating solution to drone

Because of many dead ends and a lack of available parts early in the process, implementing a solution on a drone was never realized as the method did not prove effective enough early enough to warrant the effort of implementation. This has affected the reach of the project and the end result. While testing on a test ring is a good step in the right direction the system likely has to be permanently mounted to a drone for it to be productive in the industry.

7.6 Environmentally positive maintenance with condition monitoring

Maintenance is often a costly endeavor, not only for the individual companies but also for the environment. For safety-critical operations, components are more often than not replaced or maintained at conservative time intervals to avoid damage to infrastructure or the public. With limited insight into the health state of components, there are few better and easier-to-implement ways of doing maintenance. Knowing the health state of for example drone motors will without a doubt drive the amount of wasted useful lifetime and possibly cost down, while also making the operations safer. Condition monitoring as a concept and possibly the methods described in this thesis could reveal faults that would normally not be discovered or would be discovered later using a conventional maintenance strategy.

Implementing condition-based maintenance does however require the method to be trustworthy, cheap, easy to implement, and less environmentally taxing than the alternative. Keep in mind also that the alternative does not need to be to throw the motors out when the timer runs out. Simple solutions like making arenas to sell or giving away the motors to a less safety-critical operation could be implemented. The producers could take motors in for refurbishment, and also give good

data with expected lifetimes based on maintenance intervals and the type of use for new and refurbished motors. Having other strategies for reducing waste could make an impact sooner in many industries and would also mean the gain in condition monitoring less than the alternative. Finding other alternatives to the environmental side of the problem could be quicker to implement. However, ultimately it is hard to argue the effectiveness of ”knowing” the remaining useful lifetime of the component through real-time data collection.

8

Conclusion and further work

This chapter will conclude the thesis and give closing remarks for suggested further work for similar testing and experimentation.

8.1 Conclusion

This thesis has experimented with and implemented a method for condition monitoring on brushless DC motors for use in a drone application, specifically the T-Motor MN505-s (260kv). An algorithm for finding the power of electrical and mechanical harmonics without good measurements of rotor speed has also been developed for this specific use case.

The findings point to promise in using motor current spectral analysis to find flaws in bearings on brushless DC motors in drones. Using time domain analysis on the motor current signal, one can also track the time domain features to gather insight into the state of the propeller. The approach of using motor current signals for motor and propeller diagnostics is however not without problems. Using motors with large deviations from unit to unit and temperature changes have a large impact on the readings and the features that come with them.

A flight-worthy solution to the problem is possible to create. If motor current

is to be used to an effect, a relationship between the features tracked and temperature should be established. This does also require measurements of the motor temperature while gathering data on the motor. A motor current-based condition monitoring system could, with the correct implementation be directly integrated into electronic speed controllers. This would both increase the reliability of the motors and also reduce costs for the end user of the motor by knowing more precisely the remaining useful lifetime of the motor.

8.2 Further work

Having a real to-failure test with multiple motors and good data collection would give insight into the different stages of wear on the motor over a realistic scenario. Having this data could be the key to further development. The test environment should ideally not have the propeller disturbing the same air it is pulling in. The effect of this on the measurements is unknown, but could potentially be fixed by simply testing in a larger room and flipping the test stand on its side. This could however affect the measurements by gravity no longer working axially to the propeller plane.

For further development, using a high-quality motor with a lower noise floor and a motor controller with logging capabilities and more options for control schemes could be beneficial for testing. When it comes to faults in bearings, motor current spectral analysis is a well-established method in the industry. However, for propellers, an accelerometer, vibration sensor, or acoustics might be better. Measuring with multiple sensors and measuring multiple phenomena during testing should be done to find an optimal solution. If motor current spectral analysis is to be used for condition monitoring and harmonics are the features deemed most interesting, the setup either needs very accurate rotor speed readings, or an even more refined algorithm to find the powers.

Testing hardware in the real world is a good way to investigate practical solutions to practical problems. It does have some problems though. Doing real-life testing introduces many variables that are hard to change one by one without affecting others. New development in the field of digital twins and computer modeling

of physical systems could be very useful. This could lead to having precise control of all variables, and testing different setups would then not require difficult and time-consuming hardware changes. One could also with the correct approach in a simulated test-to-failure just roll back the virtual motor to an earlier state of wear if new findings require new types of tests.

Bibliography

[1] Arunkumar, K. and Manjunath, T. (2016). A brief review/survey of vibration signal analysis in time domain. *SSRG International Journal of Electronics and Communication Engineering (SSRG-IJECE)*, 3(3).

[2] Balanda, K. P. and MacGillivray, H. L. (1988). Kurtosis: A critical review. *The American Statistician*, 42(2):111–119.

[3] Barroso-Barderas, E., Rodriguez-Sevillano, A. A., Bardera-Mora, R., Crespo-Moreno, J., and Matias-Garcia, J. C. (2022). Design of non-conventional flight control systems for bioinspired micro air vehicles. *Drones*, 6(9).

[4] Bhavsar, R. C. and Patel, R. (2013). Various techniques for condition monitoring of three phase induction motor-a review. *International Journal of Engineering Inventions*, 3(4):22–26.

[5] Chen, C.-H. and Cheng, M.-Y. (2007). A new cost effective sensorless commutation method for brushless dc motors without phase shift circuit and neutral voltage. *IEEE Transactions on Power Electronics*, 22(2):644–653.

[6] Cocconcelli, M., Bassi, L., Secchi, C., Fantuzzi, C., and Rubini, R. (2012). Mechanical systems and signal processing. *Mechanical Systems and Signal Processing*, 27:667–682.

[7] Crescentini, M., Syeda, S. F., and Gibiino, G. P. (2022). Hall-effect current

sensors: Principles of operation and implementation techniques. *IEEE Sensors Journal*, 22(11):10137–10151.

[8] Devaney, M. and Eren, L. (2004). Detecting motor bearing faults. *IEEE Instrumentation & Measurement Magazine*, 7(4):30–50.

[9] Golmakani, H. R. and Fattahipour, F. (2011). Optimal replacement policy and inspection interval for condition-based maintenance. *International Journal of Production Research*, 49(17):5153–5167.

[10] Goyal, D. and Pabla, B. S. (2015). Condition based maintenance of machine tools—a review. *CIRP Journal of Manufacturing Science and Technology*, 10:24–35.

[11] Guo, B., Song, S., Ghalambor, A., and Lin, T. R. (2014). Chapter 17 - an introduction to condition-based maintenance. In Guo, B., Song, S., Ghalambor, A., and Lin, T. R., editors, *Offshore Pipelines (Second Edition)*, pages 257–297. Gulf Professional Publishing, Boston, second edition edition.

[12] Lee, U. H., Pan, C.-W., and Rouse, E. J. (2019). Empirical characterization of a high-performance exterior-rotor type brushless dc motor and drive. In *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 8018–8025.

[Marcus10110] Marcus10110. Saleae Logic 2 Calibration Editor. `https://gist.github.com/Marcus10110/ecb1f1e894b165976791822b050b4af9`. Accessed: 2023-06-05.

[14] Papathanasopoulos, D. A., Mitronikas, E. D., Giannousakis, K. N., and Dermatas, E. S. (2020). An alternative approach for condition monitoring of brushless dc motor drives. In *2020 International Conference on Electrical Machines (ICEM)*, volume 1, pages 1280–1286.

[15] Pillay, P. and Xu, Z. (1996). Motor current signature analysis. In *IAS '96. Conference Record of the 1996 IEEE Industry Applications Conference Thirty-First IAS Annual Meeting*, volume 1, pages 587–594 vol.1.

[16] Pourpanah, F., Zhang, B., Ma, R., and Hao, Q. (2018). Anomaly detection and condition monitoring of uav motors and propellers. In *2018 IEEE SENSORS*, pages 1–4.

[17] Prajapati, A., Bechtel, J., and Ganesan, S. (2012). Condition based maintenance: a survey. *Journal of Quality in Maintenance Engineering*, 18(4):384–400.

[18] Prasad, G., Ramya, N. S., Prasad, P., and Das, G. T. R. (2012). Modelling and simulation analysis of the brushless dc motor by using matlab. *International journal of innovative technology and exploring engineering (IJITEE)*, 1(5):27–31.

[19] Ramachandran, M. and Siddique, Z. (2018). *Statistical Time Domain Feature Based Approach to Assess the Performance Degradation of Rotary Seals*, volume Volume 13: Design, Reliability, Safety, and Risk of *ASME International Mechanical Engineering Congress and Exposition*. _eprint: https://asmedigitalcollection.asme.org/IMECE/proceedings-pdf/IMECE2018/52187/V013T05A071/2504235/v013t05a071-imece2018-87857.pdf.

[20] Randall, R. B. (2010). *Vibration-based Condition Monitoring*. Wiley.

[21] Rathi, N., Ahmed, A., and Kumar, R. (2011). Comparative study of soft switching and hard switching for brushless dc motor. *International Journal of Recent Trends in Electrical and Electronics Engineering*, 1:1–5.

[22] Rausand, M., Barros, A., and Høyland, A. (2020). *System reliability theory: models, statistical methods, and applications*. Wiley series in probability and statistics. John Wiley & Sons, Inc, third edition edition.

[23] Scherz, P. and Monk, S. (2016). *Practical electronics for inventors*. McGraw-Hill Education, New York, 4th ed. edition.

[24] Shannon, C. (1949). Communication in the presence of noise. *Proceedings of the IRE*, 37(1):10–21.

[25] Stokkermans, T., Nootebos, S., and Veldhuis, L. (2019). Analysis and design of a small-scale wingtip-mounted pusher propeller.

[26] Theys, B., Dimitriadis, G., Hendrick, P., and De Schutter, J. (2016). Influence of propeller configuration on propulsion system efficiency of multi-rotor unmanned aerial vehicles. In *2016 International Conference on Unmanned Aircraft Systems (ICUAS)*, pages 195–201.

[27] Thirumarimurugan, M., Bagyalakshmi, N., and Paarkavi, P. (2016). Comparison of fault detection and isolation methods: A review. In *2016 10th International Conference on Intelligent Systems and Control (ISCO)*, pages 1–6.

[TronicsBench] TronicsBench. pigpio library. `https://abyz.me.uk/rpi/pigpio/python.html`. Accessed: 2022-12-16.

Appendix

A Flawed harmonic in Mad prop Motor 1 vs Motor 2

This appendix shows a better view of the sub-optimal pick of 4th and 5th harmonics in the Mad propeller motor 1 vs motor 2 test presented in section 6.4.1.

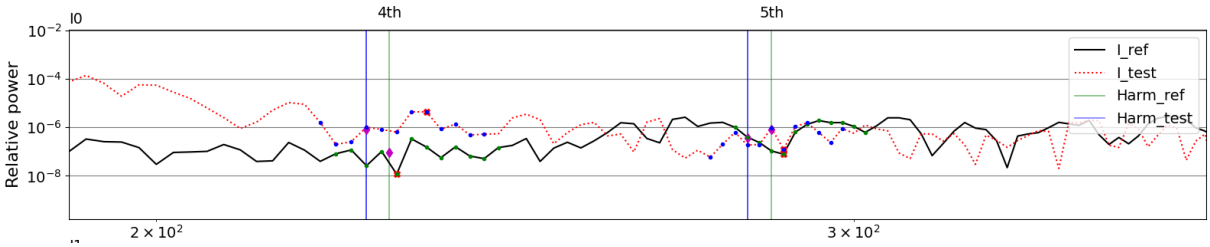


Figure 8.1: Algorithm picking sub-optimal point, resulting in a factor 10 higher difference between reference and test data.

B Example view from Logic 2 software and sensorless commutation

This appendix covers some theory regarding sensorless commutation and a view of the Logic 2 software.

For completeness, a view of a small part of the time domain signal from the logic 2 software is presented here in figure 8.2. In the figure, some areas relevant to the motor commutation steps are marked in the first phase current I0. The pink marked area to the left in the graph is the first high current drive from the ESC after this there is a floating period before the low drive phase marked in blue. Both the drive phases are 120 degrees of rotation and the floating periods are 60 degrees each, this makes for a full 360 degrees of electrical rotation shown in green in the middle of the graph.

To illustrate how the sensorless commutation works, a red field is marked on I0, and a marker pair is put down. The red field illustrates the back EMF exposure period where the motor controller does not actively drive this phase, this area is 60 degrees in length. The first part of the red area is the so-called commutation period, this is when the controller first sees the back EMF of the previous phase. To verify the motor position, the controller waits until the back EMF crosses zero. When the back EMF crosses zero the controller knows that there has to be 30 electrical degrees before the activation of that phase. This control scheme works well for a motor spinning at a higher and relatively stable RPM. If however there is a rapid change in the RPM of the motor, the controller cannot know the next optimal time for phase activation. These 30 degrees of waiting are based on at best an estimator and at worst a guess based on previous time differences from one zero crossing to another. This makes it so that such a sensorless control scheme will never be able to optimally control the motor when accelerating or at a very low rotational speed.

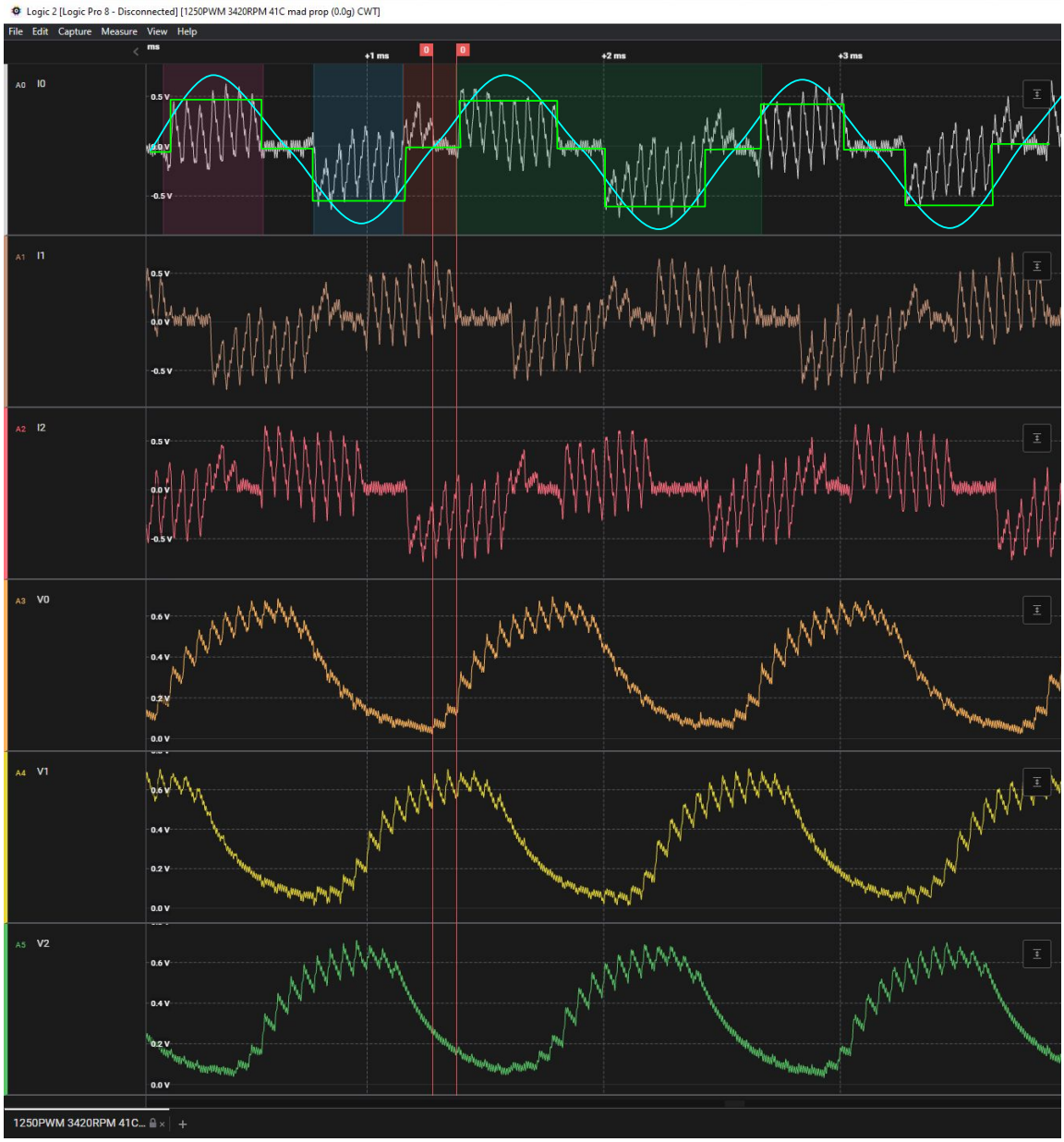
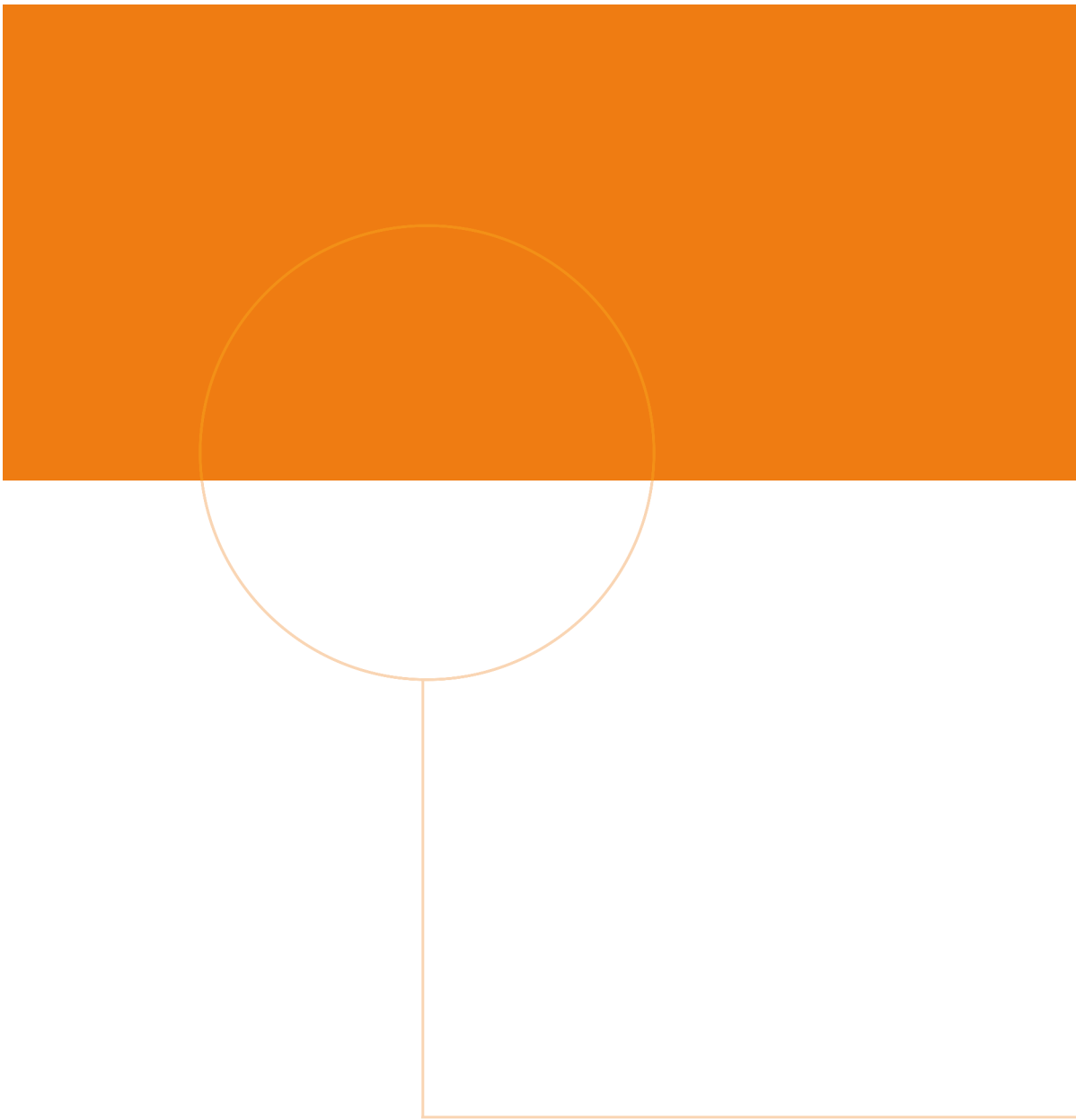


Figure 8.2: Window from Logic 2 software



NTNU

Norwegian University of
Science and Technology