Motor Control in Aerospace, optimizing availability and acoustics

Stefanos Skoulaxinos

Thesis Submitted to the University of Nottingham
for the degree of Doctor of Philosophy

Contents

Abstract

The objective of this research project was to investigate motor control methods applied to Permanent Magnet Synchronous Motors (PMSMs) for aerospace applications.

In specific this research attempted to address two key issues that are critical in aerospace. Firstly the increase in system availability in case of a resolver failure by means of applying sensorless motor control methods. Secondly the reduction of acoustic noise generated from a motor drive. Reliability, availability and acoustics are key areas in a number of industries especially aerospace.

With regards to the reliability and availability objective, a hybrid model/saliency based sensorless method was investigated that can take over motor control in case of a resolver failure.

With regards to the objective on acoustics, the research attempted firstly to address the problem of acoustic noise from High Frequency Injection (HFI). A variant of the Pseudo Random High Frequency Injection (PRHFI) algorithm was thus developed aiming to reduce the perception of acoustic noise. While investigating HFI sensorless methods and observing their acoustic effects, the most novel contribution of this research was conceived. The concept of Active Noise Cancellation/Control (ANC) by means of High Frequency Injection (HFI) was thus created, implemented and presented in this thesis.

The proposed availability and acoustic improvement algorithms were first simulated in Matlab/Modelsim and then tested on the Helicopter Electro-Mechanical Actuation System (HEMAS). The above hardware platform is a PMSM based drive used to control the swash-plate onboard a helicopter.

The reliability enhancement sensorless observer was demonstrated successfully during testing and was shown to track the motor's speed and angle.

The acoustic suppression algorithms (Pseudo Random High Frequency Injection and High Frequency Injection Active Noise Cancellation) were also demonstrated successfully on the hardware platform by means of audio capturing using microphones and analysis within Matlab.

Acknowledgements

I would like to take this opportunity to thank all the people that were involved in this project. My supervisors Professor Pat Wheeler, Dr. Gaurang Vakil and Professor Pericle Zanchetta for their guidance throughout this research. My internal and external examiners Dr. Alan Watson and Professor Dani Strickland for their valuable support and the time they invested on reviewing the thesis. Collins Aerospace for funding this research degree and specifically Mr. Ian Bedder and Dr. Andrei Dinu for their valuable advice and assistance. Finally, my fellow researcher Patrick Xie for his introduction to the HEMAS hardware platform and support.

Acronyms

AC	Alternating Current
ACE	Actuator Control Electronics
ANC	Active Noise Control/Cancellation
APU	Auxiliary Power Unit
BEMF	Back Electro Motive Force
BFM	Bus Functional Model
BPF	Band Pass Filter
BV	Blade Vortex
DAL	Design Assurance Level
DC	Direct Current
DTD	Dead Time Distortion
DSP	Digital Signal Processor/Processing
DUT	Device Under Test
EMA	Electro-Mechanical Actuator
EMI	Electro Magnetic Interference
EMF	Electro Motive Force
FOC	Field Oriented Control
FPGA	Field Programmable Gate Array
HDL	Hardware Description Language
HEMAS	Helicopter Electro-Mechanical Actuator System
HF	High Frequency
HFI	High Frequency Injection
HPF	High Pass Filter

IGBT	Insulated Gate Bipolar Transistor
IPMSM	Interior Permanent Magnet Synchronous Motor
LPF	Low Pass Filter
LSB	Least Significant Bit
MEA	More Electric Aircraft
NASA	National Aeronautics Space Administration
PMSM	Permanent Magnet Synchronous Motor
PRHFI	Pseudo Random High Frequency Injection
PWM	Pulse Width Modulation
SPMSM	Surface Mounted Permanent Magnet Synchronous Motor
RMS	Root Mean Square
RO	Read Only
RTL	Register Transfer Level
RW	Read Write
SRM	Switched Reluctance Motor
VHDL	Very High speed Description Language

Chapter 1 Introduction

1.1 Electric motor types and suitability for aerospace

A number of safety critical functions onboard an aircraft such as the control of aerodynamic surfaces have been commonly based in the past decades on hydraulic systems. However, hydraulics tend to be heavy, occupying large volume on an aircraft and lacking flexibility in terms of their operation, fault detection and fault reporting [1], [2], [3]. It is not surprising that in the past decades, hydraulic systems in aerospace tend to be replaced by electric motor drives, providing a more efficient, compact and lighter implementation.

Aiming to address concerns on sustainability and the ecological impact of fossil fuels, a converging shift is being observed with electric propulsion aiming one day to replace the currently dominating jet engine technology. This trend of increased levels of electrification, aiming for more efficient, compact and environmentally friendly applications, is known as the More Electric Aircraft (MEA) Initiative [1], [2], [3].

Electric motors belong to a group of devices called electrical machines. Electrical machines are rotating power converters and can be classified to motors and generators [4]. Motors convert electric to kinetic energy and generators convert kinetic to electric energy. An electrical machine can be used either as a motor or as a generator as long as the necessary electrical and mechanical blocks surrounding the machine are in place. Electrical machines are composed of two key parts, the stator and the rotor. The stator is a stationary hollow cylinder structure, that accommodates the rotating part of the motor named rotor (see Figure 1.1). The focus of this research is on electric motors and specifically their use in the aerospace sector.

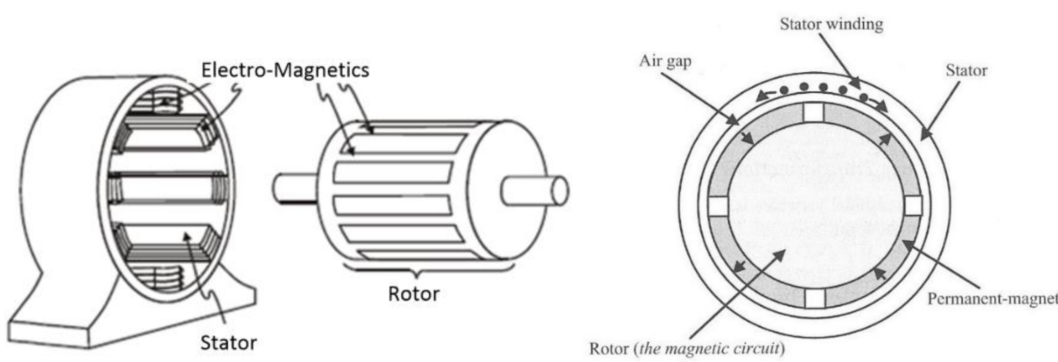


Figure 1.1 Electrical machine architectures, stator and rotor, Credit [5]

Due to the safety critical nature of many aerospace applications, stringent constraints are applied to airborne electronic hardware in general [6], [7] and to motor drives in particular with respect to reliability, weight, size, power specification, EMI, fire hazards, endurance to extreme vibration and temperature. These aerospace constraints naturally qualify certain motor types as more suitable for aerospace applications.

A wide variety of electric motor types is currently available at the designer's disposal with each type characterised by the advancements in analog, digital and material technology. Motor types can differ in a multitude of ways, with respect to their voltage supply (DC versus AC), power rating, rated speed, speed torque characteristics, winding types, number of poles, cooling method, manufacturing process and structure.

Direct Current (DC) motors establish torque and rotation with the application of a DC voltage supply at the motor's terminals [8]. The interaction of the stator/rotor magnetic fields is managed by mechanical components called brushes and a commutator assembly [9] (see Figure 1.2). DC motors need a less complicated control circuit to operate relative to AC machines, a characteristic that made their use widespread when state of the art digital controllers were not available. However they suffer from a number of disadvantages. Industrial DC motors demand high levels of maintenance, needing to be taken regularly offline, have the brushes replaced and the commutator resurfaced [9], [10], [11]. Due to the commutator assembly, DC motors also tend to be larger and heavier than an equivalent power AC machine [10]. They also tend to have higher inertia than equivalent AC machines resulting into slower speed response [10]. The mechanical commutator creates a limit on the maximum speed that can be achieved [10] while it can even create sparks while spinning [9]. All of the above characteristics make DC motors a rarely encountered machine in the safety critical aerospace sector.

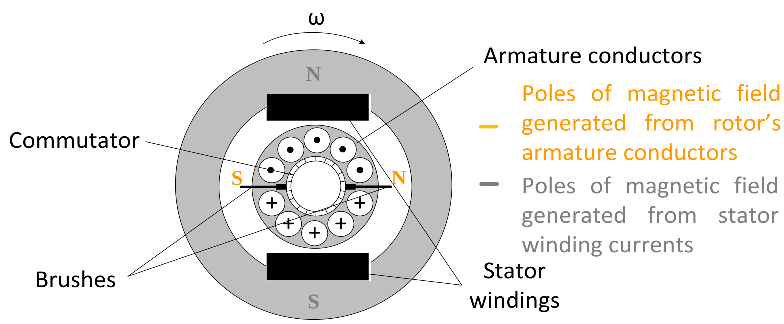


Figure 1.2 One pole pair DC motor architecture

Stepper and Switched Reluctance Motors (SRMs) is a group of electrical machines that initiate torque and rotation using reluctance torque. They both have projecting poles and share the characteristic of having different number of poles in stator versus rotor. Stepper motors and SRMs are considered economical to manufacture, maintenance free and very reliable machines. The rotor of a SRM is formed by steel laminations while the rotor of stepper motor can be either formed by steel laminations or permanent magnets depending on the stepper motor type. However stepper motors can only produce sufficient torque at low speed [12], they have low power density and their torque to inertia ratio is low so they accelerate slowly. A limitation for both stepper motors and SRMs is torque ripple [13], [14] and resultant acoustic noise.

Alternating Current (AC) motors can be classified into asynchronous machines (induction motors) and synchronous machines (Brushless DC Motors, PMSMs) [5]. Both types of AC machines are considered economical, relatively straightforward to manufacture and are practically maintenance free [9]. They are more lightweight and efficient than equivalent power DC motors [9]. Induction motors however require excitation currents to flow through the rotor windings thus need to be of bigger physical size when compared to equivalent permanent magnet motors to prevent overheating [10]. Furthermore brushless DC motors produce higher torque ripple than PMSMs [15], [16]. In conclusion, considering the high power density of PMSMs, their dynamic performance and the lack of maintenance needs, they are considered an excellent option for high performance safety critical motor control applications in the aerospace sector. PMSMs have been used for the control of aerodynamic surfaces onboard an aircraft and therefore form the focus of this research work.

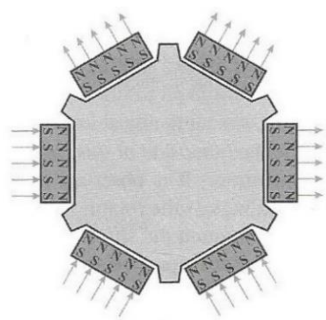


Figure 1.3 One pole pair Surface Permanent Magnet Synchronous Motor (SPMSM), Credit [5]

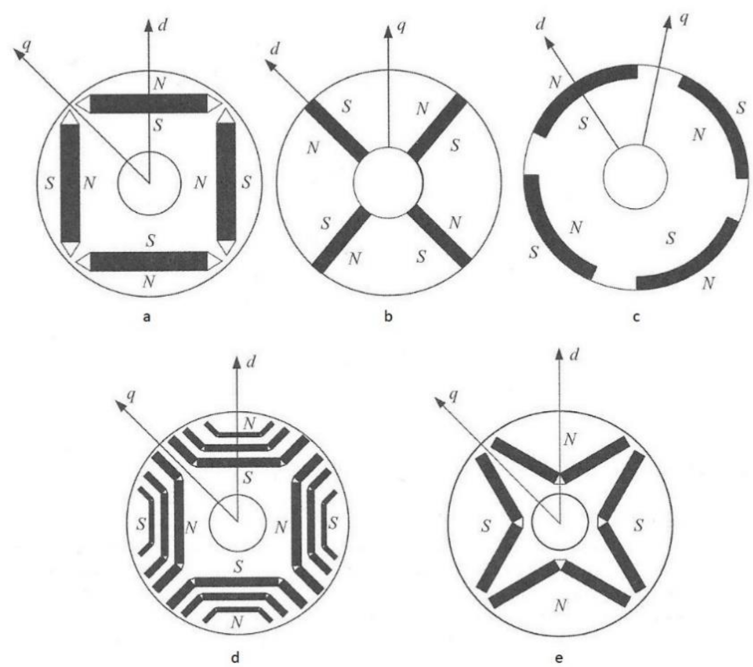


Figure 1.4 two pole pair Interior mounted Permanent Magnet Synchronous Motors (IPMSMs): 4a) radially magnetized, 4b) tangentially magnetized, 4c) inset magnetized, 4d) multi layered magnetized, 4e) V-shape, Credit [5]

Permanent Magnet Synchronous Motors are manufactured by embedding a permanent magnet to the rotor and a set of conductors to the stator known as windings. There are primarily two types of PMSMs depending on the topology of the permanent magnet, Surface mounted Permanent Magnet Synchronous Motors (SPMSMs) (Figure 1.3) and Interior mounted Permanent Magnet Synchronous Motors (IPMSMs) (Figure 1.4) [4], [17], [5].

The location of the permanent magnet within the rotor affects the overall magnetic circuit and the characteristics of the machine. Each of the PMSM topologies shown in Figure 1.3 and 1.4 has therefore its advantages, disadvantages and higher suitability for specific applications as will be analysed below. SPMSMs for example are known for their high power density while most IPMSMs are considered attractive for using their reluctance torque above base speed, a method widely used in traction applications in the automotive sector [5].

Specifically, radially magnetized IPMSMs shown in Figure 1.4.a exhibit low saliency ratio L_q/L_d and therefore modest reluctance torque [5]. Tangentially magnetized IPMSMs shown in Figure 1.4.b, unlike other IPMSMs exhibit higher d-axis inductance L_d than q-axis inductance L_q [5] and therefore their reluctance torque cannot be used towards enhancing performance. Inset IPMSMs illustrated in Figure 1.4.c have the advantage of operating in an extended torque speed range and have been used in commercial hybrid vehicles. The multilayered IPMSM topology shown in Figure 1.4.d exhibits high saliency ratio L_q/L_d therefore supporting high reluctance torque. However, their design is considered more complex than other IPMSMs [5]. Finally the V-shaped topology shown in Figure 1.4.e, exhibits wider constant power speed range than other IPMSM types and has enjoyed commercial applications in hybrid vehicles [5].

Motor type	Pros	Cons
1. Brushed DC motor	• Straightforward control circuit (DC voltage supply). • Economical for very low-cost low-performance motors (toys).	• Complex and expensive to manufacture for high end industrial DC applications. • High maintenance. • Larger and heavier than equivalent AC machines. • Higher Inertia and Lower maximum speed than equivalent AC machines. • Can create sparks.
2. Stepper motor and SRM	• Straightforward and economical to manufacture. • Maintenance free and robust.	• Torque ripple and acoustic noise at high speed. • Stepper motors efficient at low speed only, slow acceleration
3. AC Motor		
3.1 Asynchronous Machine		
3.1.1 Induction Motors	• Straightforward and economical to manufacture. • Maintenance free and robust. • Higher efficiency and power density than DC machines.	• Compared to PMSMs, excitation currents through rotor windings reduce efficiency add thermal considerations and increase motor size.

3.2 Synchronous Machine		
3.2.1 Brushless DC Motor	• Straightforward and economical to manufacture.	• Higher torque ripple and reduced efficiency when compared to PMSMs.
3.2.2 PMSM	• Straightforward and economical to manufacture. • Maintenance free and robust. • Higher efficiency and power density than DC machines and induction motors.	• Cost of permanent magnet used to be a disadvantage, this cost has been reduced greatly. • Risk of demagnetization in early products, not a risk in modern systems.
3.2.2.1 SPMSM	• Superior power density than DC, Induction, Brushless DC, stepper motors, SRMs, and IPMSMs.	• Torque reduced when used above base speed when compared to IPMSMs.
3.2.2.2 IPMSM	• Superior power density than DC, Induction and Brushless DC Motors. • High torque can be maintained above base speed using reluctance torque (ideal for traction applications).	• Not as power dense as SPMSMs when used up to base speed.

Table 1.1. Comparison of electric motors

Motor type	**Suitability for aerospace**
1. Brushed DC motor	Not suitable for a aerospace applications. They require high maintenance, create sparks during rotation, are of low efficiency and low power density.
2. Stepper motor and SRM	SRMs have some usage in aerospace, they are maintenance free, they are considered very robust however they are not as power dense and efficient as PMSMs and are known to experience high torque ripple and acoustic noise. Stepper motors have very narrow and limited applications due to speed torque characteristics.
3. AC Motor	
3.1 Asynchronous Machine	
3.1.1 Induction Motor	Induction motors have usage in aerospace, they are maintenance free, straightforward to manufacture and more power dense than DC machines. However their power density and efficiency is lower than that of PMSMs.
3.2 Synchronous Machine	
3.2.1 Brushless DC Motor	Brushless DC motors have some usage in aerospace, they are maintenance free, however they are known to experience high torque ripple.
3.2.2 PMSM	Surface mounted PMSMs (SPMSMs) are considered an excellent choice for high performance, high power density aerospace applications. IPMSMs have a more narrow application range in aerospace as their reluctance torque make them more suitable for traction applications.

Table 1.2. Electric motor types and suitability for aerospace

1.2 Magnetic field and rotation in electric motors

The fundamental principle behind the energy conversion in electric motors is the interaction between the stator's and rotor's magnetic field an effect known as magnetic field coupling [4], [9]. A magnetic field can be created either by means of introducing a current flow through a conductor or naturally around a permanent magnet (see figure 1.5). A key parameter of a magnetic field is its magnetic flux. Magnetic flux lines—a visual representation of flux—are shapes that would be formed if steel pins were to be placed within a magnetic field. Flux lines show direction of flux as well as the intensity of the magnetic field. The closer the flux lines, the higher the flux density and the intensity of the field. Flux density B is a physical measure of magnetic field intensity i.e. how close the flux lines are situated to each other and the physical unit is a Tesla (T). Flux lines indicate the location of the poles of a magnetic field namely North (N) and South (S) (Figure 1.5).

It can be shown that as magnetic fields interact, Force can be produced. Using Lorentz law [4], the Force F that is experienced by a moving charged particle of electric charge q, when travelling at speed v through a magnetic field of Flux density B is:

$$F = qvB \qquad \text{Eq 1.1}$$

Starting from Lorentz law defined in Eq 1.1, it is possible to substitute qv with $q\ {}^{s}/_{t}$ as the speed v equals to the ratio of travelled distance s in time t over this time. Additionally, it is possible to substitute ${}^{q}/_{t}$ with I, as electric current is defined as the rate of charge movement over time. The above two substitutions result into Lorentz law being re-formulated to the familiar equation Eq 1.2 shown below [8]:

$$F = BIl \qquad \text{Eq 1.2}$$

where F is the Force experienced by a conductor of length l, carrying current I and exposed to a magnetic field of flux density B.

Aiming to explain this field interaction further, let’s consider the stator and rotor magnetic fields are generated by embedding a permanent magnet or applying current through windings. This will result into the stator and rotor developing magnetic fields each with N and S magnetic poles. Matching stator/rotor poles (N / N and S / S) will attempt to distance from each other and opposite stator/rotor poles (N / S) will attract each other. This synchronism and interaction of rotating stator and rotor fields is the key mechanism to exert force and establish rotation in Permanent Magnet Synchronous Motors. Figure 1.6 illustrates this field coupling and rotation within a PMSM. Note that both the rotor and stator experience equal and opposite in direction force. This is the reason why the former is designed to rotate freely and the latter is typically fixed on a surface to remain stationary.

This synchronism of stator/rotor magnetic fields, where one field is oriented in vectorial relation with the other and produced by applying appropriate voltages to the stator terminals is known as Field Oriented Control (FOC). However, in order to achieve this field orientation and establish stable control, the motor controller needs to be informed of the rotor’s angle. The rotor position is for this reason a very significant piece of information for the controller. Position encoders known as resolvers are therefore commonly embedded within PMSMs to achieve this.

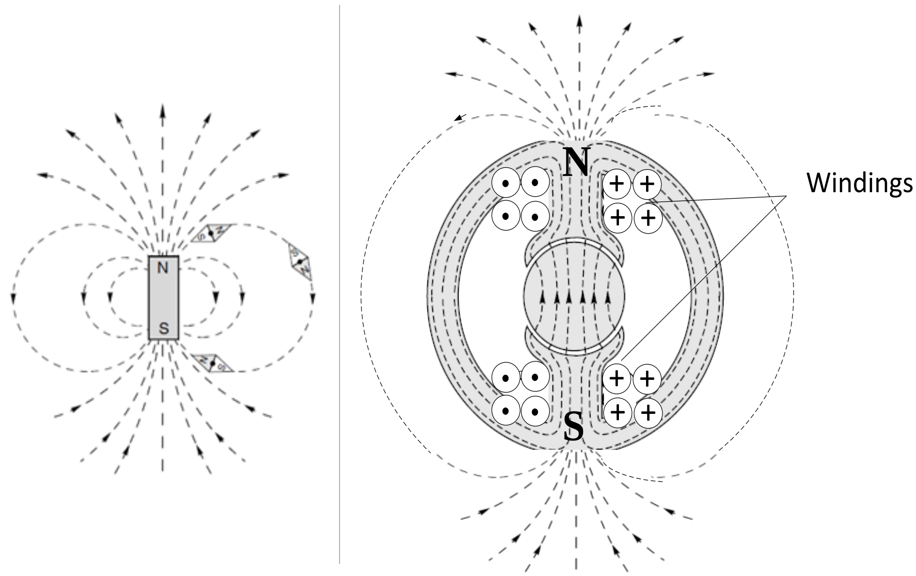


Figure 1.5 Magnetic flux lines and magnetic poles: Permanent magnet (Left), current carrying windings (Right), Credit [9]

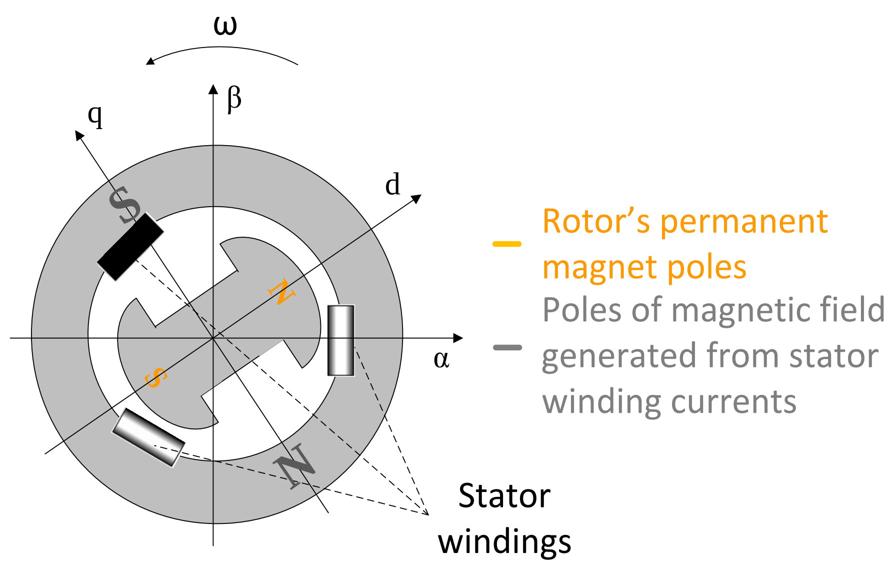


Figure 1.6 Field coupling for one pole pair 3 phase AC PMSM

1.3 EMAs and the HEMAS platform

Electro-Mechanical Actuators (EMAs) are systems that utilise electric motors and convert their rotary motion into linear movement. There is a clear trend of EMAs replacing heavier hydraulics [1], [2], [3]. especially in newer lightweight composite based planes where efficiency is key. PMSM based EMAs can thus be commonly found in safety critical aerospace applications such as the control of aerodynamic surfaces (rudder, elevator, aileron, trimming horizontal stabilizer).

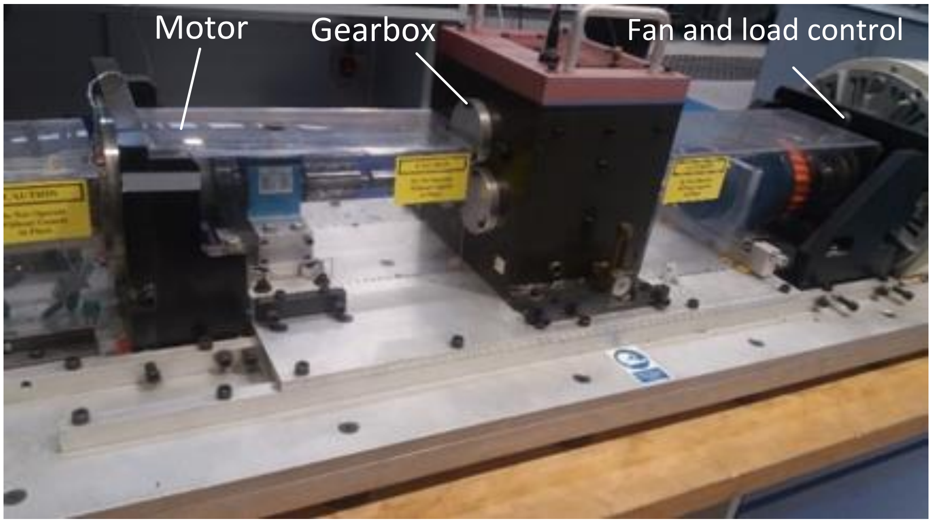


Figure 1.7 The HEMAS hardware test platform

The EMA that was used for hardware testing in this research is the Helicopter Electro-Mechanical Actuation System (HEMAS) [18], [19], [20]. The HEMAS system in an SPMSM based drive located in correspondence of the swash-plate of a helicopter. Chapter 3 of this thesis provides a more detailed description of the HEMAS platform and the test rig used in the research (See Figure 1.7).

Using the HEMAS system as a test platform is an excellent choice considering that many aerospace applications will share many fundamental motor technology similarities.

1.4 Current state of the art, research motivation and objectives

The use of PMSMs has become increasingly widespread in the past years due to their high performance and power density. However one key characteristic of PMSMs is the need for the motor controller to be aware of the rotor angle. This need has been typically accommodated with the use of a resolver embedded in the motor. However, as will be detailed in chapter 2, resolvers can fail due to extreme temperature and vibration resulting into reduction of system availability. The safety critical nature of aerospace applications and the expectation for high levels of reliability especially for DAL A/B designs [6] make the weakness of resolver failures a significant drawback for conventional PMSM drive solutions. To increase system reliability, this research investigates the use of sensorless methods to enable motor control in the case of a resolver failure.

Another key aspect investigated in this research is acoustic noise originating from a motor drive. Acoustics is of critical significance in aerospace for the comfort of passengers onboard commercial aircrafts and the safety of personnel onboard military planes. This research therefore investigates and proposes a number of ways that acoustic noise can be suppressed and reduced.

1.5 Thesis organisation

This thesis is organized in the following chapters. Chapters 2 and 3 provide a background on the two main areas investigated in this research, system availability and acoustics. With respect to availability, Chapter 2 presents reliability considerations for PMSMs and specifically the mechanism of resolver failures. Aiming to increase availability, the chapter introduces sensorless methods that are proposed to take over control in case of a resolver failure. It provides a survey and comparison of different sensorless methods, illustrating key advantages and disadvantages of each method. With respect to acoustics, Chapter 3 analyses the types of acoustic noise typically encountered onboard an aircraft and then outlines two proposed algorithms to improve acoustics.

Chapter 4 presents the hardware platform and test setup for the experimental phase of this research. At first it provides an insight to the Helicopter Electro-Mechanical Actuation System (HEMAS). It then introduces the software/hardware partitions of the system and the test methodology adopted during this research.

Chapter 5 details the proposed model based sensorless observer towards increasing motor drive and system availability at medium to high rotational speed. The chapter presents an analysis of the algorithm, along with simulation and hardware test results on the HEMAS platform.

Chapter 6 presents the proposed saliency based sensorless method towards increasing system availability at standstill or low speed. It presents an analysis of the method along with simulation and hardware test results on the HEMAS platform.

Chapter 7 details the proposed Pseudo Random High Frequency Injection (PRHFI) algorithm aiming to suppress acoustic noise from HFI. It details the algorithm and then presents simulation and hardware test results on the HEMAS platform.

Chapter 8 presents the most innovative and novel method introduced in this thesis. It was named in this research, Active Noise Cancellation (ANC) by means of High Frequency Injection (HFI). It provides a background on ANC systems in general, followed by the description of the proposed method along with simulation and hardware test results on the HEMAS platform.

Chapter 9 is the final chapter of the thesis and contains the conclusions and direction towards future work related to this research project.

Chapter 2 System Availability for Motor Drives

2.1 Introduction

Reliability and availability is an area of great significance in aerospace. This chapter first provides a brief background on availability considerations for Motor Drives and specifically reliability concerns for PMSM drives due to resolver failures. The architecture of a resolver and its failure mechanism are therefore presented. Aiming to enhance availability, sensorless control methods are proposed to take over motor control in case of a resolver failure. A survey of sensorless methods is therefore performed showing strengths and weaknesses of each group of algorithms, followed by a brief analysis of how sensorless methods can be used towards increasing system availability.

2.2 System availability

2.2.1 Resolver architecture and failure mechanism

The controller of a PMSM drive needs to be continuously informed of the rotor angle to ensure efficient and stable motor control. The most commonly used method to achieve this is by embedding a resolver to the motor.

Motor resolvers are electro-mechanical components that monitor rotation and provide electrical signals conveying the position of a rotor. The construction of a resolver is based on three windings, a primary winding and two secondary windings [21], [22]. The primary winding is embedded within the rotor and the two secondary windings are placed on the stator at quadrature angle of 90^0 (see Figure 2.1). The primary winding is fed with an excitation signal typically within the 10 kHz frequency range. The resolver design functions as a transformer and depending on the angle of the rotor, the excitation signal will appear to each of the output windings (see Figure 2.1 and 2.2). The feedback from the two output windings is known as the resolver feedback sine and cosine signals.

$$Resolver\_cosine\_feedback = sin(\omega_{exc}) * cos(\theta) \qquad \text{Eq 2.1}$$

$$Resolver\_sine\_feedback = sin(\omega_{exc}) * sin(\theta) \qquad \text{Eq 2.2}$$

where θ is the angle of the rotor modulated to the excitation signal frequency ω_{exc}.

Note that the modulated sine and cosine feedbacks contain information on the rotor angle that can be demodulated at the controller.

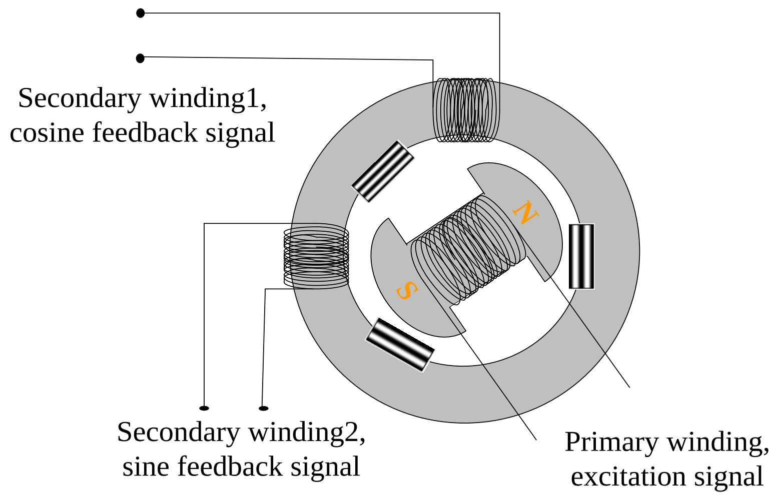


Figure 2.1 Resolver architecture

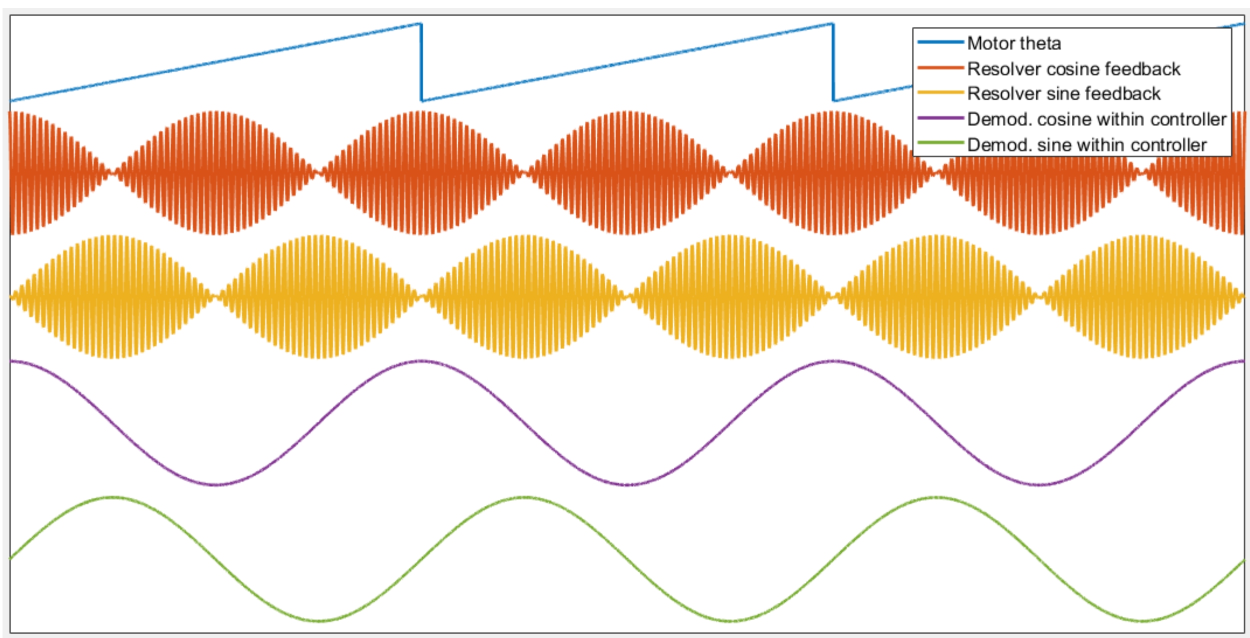


Figure 2.2 Resolver sine/cosine feedback vs angle theta

*Note that a Matlab model *.m file was written to generate the resolver feedback signals shown in figure 2.2 please refer to Appendix B1.1 of this thesis to obtain the Matlab source code.*

While resolvers provide a viable solution to convey the electrical angle information, they are naturally sensitive devices. They can fail due to extreme temperature and vibration conditions, leading to inability of the controller to drive the motor and reduction in system availability.

The underlying mechanism of a resolver failure due to temperature and vibration is detailed below:

1. Resolvers are transformers and therefore need current flow to function. Conductors carrying the current to the resolver within the motor and the surrounding insulating material however respond to temperature. Extreme temperatures or fast changes in temperatures cause these conductors and their neighboring insulation to expand and contract. The factor that amplifies this problem is that the thermal expansion coefficients of the conductors and neighboring insulation differs substantially causing them to expand and contract by a different degree. This unbalanced movement results into extreme mechanical stress that can eventually cause open or short circuits to the conductors. This in turn results to the resolver not being fed with the needed current and to stop functioning.
2. The primary winding block of the resolver is embedded on the rotor. Vibration and mechanical shock can result into the resolver rotating circuit slipping and becoming detached from the main rotor or suffer an impact to its electrical connection integrity. In either case this results into the resolver stop functioning.

Aiming to prevent resolver failures decreasing system availability, this research proposes applying sensorless methods that can take over motor control in the case of a resolver failure. The section below presents a survey of sensorless methods aiming to identify a suitable sensorless algorithm to be implemented and evaluated in this research.

2.2.2 Survey of sensorless methods

In view of the disadvantages encountered in resolver based motor control designs, sensorless motor control has been an area of research in the past decades. Sensorless position calculation involves calculating the angle of a motor without the use of an encoder but by means of monitoring feedback signals such as motor voltages and currents.

Sensorless motor control algorithms for PMSMs can be classified into two main groups, model based methods and saliency based methods. The former group involves utilising motor feedback signals during rotation to establish the rotor angle, considering the equivalent circuit model of the motor. The latter group takes advantage of anisotropies in PMSMs by means of injecting High Frequency (HF) carriers and analysing their effect to the motor’s feedback signals. Model based methods are considered suitable for medium to high speed motor rotation as they rely on feedback signals while the motor is spinning while saliency methods are typically applied at standstill and low speed. Below is a survey of sensorless motor control methods for PMSMs:

2.2.2.1 Model based methods

Open Loop Back Electromotive Force (EMF) estimator): The method calculates the rotor angle by integrating the back EMF of a spinning motor. It is straightforward to implement however being open loop the method leads to accumulative errors [23] due to sampling noise and DC offset while it is also vulnerable to motor parameter variation [23]. Considering the above, this method is not widely used.

Closed loop and observer methods: In order to mitigate against noise and enable self-corrective capability closed loop methods and observers were introduced:

Extended Kalman Filtering observer (EKF) and variants: The EKF is a widely used observer [24]. The method is highly effective in processing a noisy source contaminated with random interference. Limitations include the difficulty to track fast changes in motor speed and tuning complexity. To remedy the former the Sub-optimal Fading Extend Kalman Filter (SFEKF) variant was proposed [25]. To remedy the latter, the Unscented Kalman Filter (UKF) [26] and Cubature Kalman Filter (CKF) [27] were proposed. The EKF observer is considered more complex to implement than the MRAS and SMO methods.

MRAS Observer: The Model Reference Adaptive System (MRAS) observer is a widely used method relying on the application of the motor equivalent circuit to two parallel models, the reference model and the adaptive model thus the name. The accuracy of this observer is susceptible to motor parameter variation. Variants of the method to mitigate this limitation have been proposed in [28] where the method is less parameter sensitive.

Sliding Mode Observer (SMO): The Sliding Mode Observer (SMO) is a method designed to extract the back EMF voltage of a spinning motor and from this calculate the motor speed and angle [29]. The SMO method is less complex to implement than the EKF and is not impacted by motor parameter variation as is the MRAS. However, the method is known to introduce jittery and noisy position estimation and motor control. Variants of the method have been proposed to mitigate the jittery performance including hybrid methods marrying the best characteristics of different methods such as the EKF/SMO.

Flux linkage Observer*:* The method calculates the rotor position by means of estimating the rotor flux [30]. The observer has a number of limitations most importantly the difficulty to accurately estimate the rotor flux. Variants of the method attempting to remedy this have been proposed including the Second Order Integral Flux Observer (SOIFO) [31]. This method however is not as widely used as the EKF, MRAS and SMO.

Fuzzy Logic and Neural Networks*:* The application of Neural Networks (NN), Fuzzy Logic (FL), and Fuzzy Neural Network (FNN) in the area of sensorless motor control has also been an area of active research [32]. A number of publications propose algorithms based on the above artificial intelligence methodologies towards the calculation of motor speed and position. Such methods are considered suitable for cases when the system is difficult to model accurately due to its complexity or non-linearity for example when attempting to compensate parameter variation.

Third harmonic detection*:* This method is a variant of the flux linkage calculator. Specifically the third harmonic flux linkage is estimated by means of third harmonic back EMF integration. This method was introduced by Moreira and Lipo in [33]. The method enjoys a number of advantages over a number of conventional model based sensorless methods including not being affected by parameter variation, and PWM noise [34].

2.2.2.2 Saliency based methods

High Frequency (HF) signal injection*:* To enable sensorless motor control at zero or low speed, High Frequency signal injection was introduced. The method involves superimposing a high frequency voltage carrier to the fundamental frequency of the motor applied either at the stationary or synchronous frames and then observing the resulting motor current feedbacks. If the motor is prominently salient i.e. Ld < Lq as is the case for an IPMSM or it becomes salient due to stator iron saturation by injecting in the direct axis of an SPMSM, the feedback currents will convey information on the position of the rotor.

Sinewave injection (rotating/pulsating)*:* The sinusoidal rotating signal injection method involves adding a carrier signal to the alpha and beta stationary frame. Disadvantages of the method involve lack of dynamic performance due to delays associated with LPF Digital Signal Processing needed to demodulate the angle and torque ripple. Aiming to reduce the torque ripple of rotating sinusoidal method, the pulsating sinewave injection in the d synchronous axis was proposed.

Square wave injection (rotating / pulsating): The inherent delay of sinusoidal injection involving the needed low pass filters to the feedback currents, can result into low dynamic performance. To mitigate this, square signal injection was introduced. This permits higher injection frequency and more dynamic performance. Similarly, it can be applied to the stationary frame and for improved torque ripple in the direct axis synchronous frame in form of pulsating square wave injection.

Acoustic noise reduction in HF injection*:* In order to reduce the acoustic noise caused by HF injection, one solution proposed was decreasing the amplitude of the injected signal [35]. However this also reduces the SNR and accuracy of the position. Another solution which is investigated in this research is injecting pseudo random frequency signal.

Inductance variation methods (INFORM): This method takes advantage of the changing magnetic conductivities in the d and q synchronous axis as the motor rotates [36]. It involves injecting a high frequency test voltage resulting into the transient inductance dominating over the voltage drop from the stator resistance. A variant of the method is named Indirect Flux detection by On-line Reactance Measurements (INFORM).

Current slope measurement method (high speed sensor): The current slope measurement method involves utilising the saliency of a PMSM and detecting the rotor position by measuring the slope of the motor currents. To do so high-speed current sensors are needed and precise sampling of the current waveform to be performed in synchronism with the PWM switching.

Current slope measurement method (Rogowski coil method): Utilising a high-speed current sensor and sampling to establish the current derivative (2.2.1) is a process where sampling needs to be carefully timed. A better method is in fact to utilise specialized hardware that measures the current derivative named Rogowski coils [37] and thus decode the rotor angle.

2.2.2.3 Open Loop methods

The method is sometimes used for start-up sensorless conditions when the position of the rotor can remain unknown until the motor starts spinning at which point a model-based method can be used. It involves applying a sinusoidal voltage and expect the motor to rotate at this frequency. The method although straightforward to implement, is very sensitive to load disturbance variations and provides little visibility of the motor's state.

2.2.2.4 Hybrid sensorless methods

Hybrid methods are based on the concept that more than one sensorless methods can be combined to run simultaneously or enabled/disabled over specific points in time. For example a number of hybrid methods have been introduced to support full speed range in sensorless mode [38] as model based methods only function when the motor is spinning. Other hybrid sensorless methods were introduced to join best characteristics of different algorithms [39]. Hybrid sensorless methods are considered a very attractive route one that is adopted and proposed in this research.

2.2.2.5 Comparison of sensorless methods

There is a rich variety of model-based methods each with varying implementation complexity, accuracy and vulnerability to motor parameter variation. Model based methods can only function when the motor is spinning.

Saliency based methods can function both at standstill and while the motor is spinning. They utilise the saliency that either exists naturally by design for some motors or induced by means of stator iron saturation. There are different methods and flavours under this category such as High Frequency Injection (square/sinusoidal, stationary/synchronous frames) and current slope measurement methods. High Frequency Injection can contributes to torque pulsation and acoustic noise and requires additional power that is not used towards spinning of the motor.

Open loop methods tend to be used in applications where system characteristics such as the load are predictable and also as a start-up method for a model based algorithm.

Hybrid methods intend to marry best characteristics of different algorithms and support features that would not be available only with one algorithm.

The advantages and disadvantages of each sensorless method can be summarized in Table 2.1.

Sensorless PMSM method and performance summary	Standstill/low	High speed	Signal noise	Parameter	Accuracy, torque	Dynamic performance	Method maturity
Model based methods							
Open loop back EMF estimator	0	1	0	1	1	1	1
Closed loop methods & observers							
EKF	0	2	2	1	1	1	2
EKF Variations (UKF, CKF, SFEKF)	0	2	2	1	1	1	1
MRAS observer	0	2	1	0	1	1	2
SMO	0	2	1	2	0	1	2
Flux linkage Observer	0	2	1	1	0	0	1
Fuzzy Logic & Neural Networks	0	2	1	1	1	1	1
Third harmonic Detection	0	2	1	2	1	1	1
Saliency based methods							
High Frequency signal Injection (HFI)							
Sinusoidal signal injection							
Stationary Frame	2	1	1	1	0	1	1
Synchronous frame	2	1	1	1	2	1	2
Square signal injection							
Stationary frame	2	1	1	1	0	2	1
Synchronous frame	2	1	1	1	2	2	2
Acoustic suppression (Reduced amplitude)	1	0	0	1	0	1	0
Acoustic suppression (Random freq)	2	1	1	1	1	1	1
Inductance variation method (INFORM)	2	1	1	1	1	1	1
Current slope measurement method							
High speed current sensor method	2	1	1	1	1	1	1
Rogowski coil	2	1	1	1	1	1	1
Open loop methods	0	0	0	1	0	0	2
Hybrid methods							
EKF and SMO	0	1	2	2	1	1	1
HF signal injection and Kalman Filtering	2	1	2	1	1	1	1
HF signal injection and SMO	2	1	1	2	1	1	1

Rating convention:
2 Excellent optimal performance
1 Good / Average performance
0 Poor / Inadequate performance

Table 2.1. Sensorless motor control methods

Justification of the rating in Table 2.1:

Standstill/low speed: Model based methods can only function when the motor is spinning as they operate by processing phase currents and phase voltages. They are therefore assigned to the rating of 0 for standstill and low speed. On the contrary, saliency-based methods are optimized for standstill thus the optimal rating of 2. The reduced amplitude acoustic suppression method is rated 1 as the approach comes at the price of reduced sensorless accuracy [35]. Open Loop methods do not estimate the rotor position and are therefore rated as 0. Finally, hybrid algorithms are assigned with ratings of the methods they incorporate.

High speed: Model-based methods are rated optimal as the physical values they measure/estimate such as Back EMF and phase currents have high values at high-speed enabling high levels of accuracy. The Open Loop Back EMF Method is considered of low performance [23] thus the assignment of rating 1. Saliency based methods function at high speed but with some limitations such as the High Frequency carrier producing less rotor angle estimation samples within an electrical period at high speed and inaccuracy of calculations as the back EMF is not negligible.

Signal Noise: EKF and its hybrid variants are known to be effective on suppressing signal noise [24] and are therefore rated with the optimal rating of 2.

Parameter variation: The Sliding Mode Observer (SMO) method is known to be highly effective towards suppressing motor parameter variations [29] therefore the optimal rating assigned.

Accuracy/torque ripple: The EKF is rated optimal for torque ripple as signal noise does not impact extensively the estimated angle [24] therefore resulting into stable I_q and reduced torque ripple. HFI in the synchronous frame (d axis) can reduce torque pulsations and therefore rated optimally in Table 2.1.

Dynamic performance: Square wave HFI is rated optimally with respect to dynamic performance as it can reach higher frequencies than sinusoidal injection and therefore obtain more saliency related rotor angle points within an electrical period allowing for fast changes in speed.

Method maturity: Model based sensorless methods that have been widely used include EKF [24], MRAS Observer [28] and SMO [29] therefore the optimal rating of 2. Open Loop methods are also used in applications when the loading characteristics of a motor drive are predictable, while synchronous HFI is a favorable saliency method aiming to minimize torque ripple.

2.2.3 Increasing system availability using sensorless methods

This chapter aims to introduce the reliability problem associated with motor resolvers and how sensorless methods can be used to address this limitation. Mazzoleni et al. and Di Rito et al. [40], [41] quantify the probability of resolver failures in the range of 10^{-6} to 10^{-7} per flight hour. They conclude that a simplex resolver design is not reliable enough to meet the safety goals for flight control applications. Past research [42] has shown that a model based sensorless method can be used to monitor the resolver interface. However [42] provides no failure detection and no increase in availability if the failure takes place at low speed or at standstill. The research presented in this thesis attempts to provide an increase in system availability at any speed using a novel hybrid sensorless method able to monitor and enable motor control throughout the motor's speed. The proposed hybrid model/saliency based method are therefore presented in detail in chapters 5 and 6 of this thesis.

2.3 Conclusions

This chapter provided a background on reliability considerations for PMSMs with respect to resolver failures. Aiming to enhance availability in the case of a resolver failure, sensorless methods were presented. A survey of sensorless algorithms was performed detailing advantages and disadvantages of each method and the concept of using sensorless control to increase availability was introduced. The detailed analytical and test results of the sensorless methods used in this research are presented in chapters 5 and 6 of this thesis.

Chapter 3 Aircraft acoustics

3.1 Introduction

Aircraft acoustics is a topic of active research and development gathering significant interest in the past decades. It impacts a number of areas including the comfort of passengers, the noise emitted to residences near an airport and for military applications, the safety of the personnel onboard.

This chapter first presents and analyses acoustic noise that is typically generated from an aircraft and then introduces two proposed algorithms aiming to reduce acoustic noise:

- A variant of the Pseudo Random High Frequency Injection (PRHFI) algorithm suppressing the acoustic noise perception from High Frequency Injection (HFI).
- The novel method of Active Noise Cancellation (ANC) by means of High Frequency Injection (HFI).

While the details of proposed algorithms are presented in chapters 7 and 8 of this thesis, this chapter serves to provide the background setting and motivation for the development of these methods.

3.2 The science of acoustics

Acoustics is one of the oldest branches of Physics [43]. It is naturally divided into four areas listed below [43]:

- The mechanism of sound generation i.e. the phenomenon initiated by the vibration of an object within an elastic medium.
- The propagation of this vibratory energy through an elastic medium in the form of waves. The molecules of a medium can be perceived in the context of acoustics as particles interconnected by springs (bonds between molecules). The initially vibrating object initiates vibration of the molecules surrounding it, which in turn transfer this vibration into other surrounding molecules causing a wave of varying pressure/compression travelling through an elastic medium until the energy is dissipated. The medium through which sound propagates can be a gas, a liquid, or a solid object.
- The physical reception of this vibratory energy (variations of pressure) from a receiver using an ear or a microphone device.
- The psychological perception of noise. In other words whether a noise is usually perceived as pleasant or not by a receiver.

Each of the above areas of acoustics is of interest in this research from how noise is generated onboard an aircraft, how it propagates and interacts with various subsystems and how it is perceived from the receiver.

One key aspect of acoustics analysis involves the sound velocity through an elastic medium [43]. Travelling of sound relies on the elasticity of the medium i.e. the strength of the bond between molecules that acts like a spring. The speed of sound has been investigated by a number of scientists including Newton and Laplace. It is expressed in the below equation provided by Newton [43]:

$$c = \sqrt{\frac{\mathrm{E}}{\rho}} \qquad \text{Eq 3.1}$$

where c is the speed of propagation of sound, E is the medium's modulus elasticity and ρ is the medium's density.

Sound propagates faster in liquids than in gases and faster in solids than in liquids. This is because the increase in medium elasticity is greater than the increase in medium density when comparing solids liquids and gases.

For liquids and gases the speed of sound is also expressed by what is known as the Newton Laplace equation:

$$c = \sqrt{\frac{\gamma * P}{\rho}} \qquad \text{Eq 3.2}$$

where γ is the Laplace correction having the value of 1.4, P is the pressure and ρ the medium's density.

Environmental parameters such as temperature and humidity affect the speed of sound (the former having a stronger effect that the latter) and they feed indirectly into equation 3.1 [43]. Temperature is inversely proportional to the medium density, so increase in temperature results into the decrease of density and therefore increase of the speed [43]. Another formula illustrating this relationship is shown below in equation 3.3 [43]:

$$velocity\ of\ sound\ versus\ Temperature\ = 331.2\sqrt{1 + \frac{T}{273}} \qquad \text{Eq 3.3}$$

where T is temperature in degrees Centigrade and the coefficient of 331.2 is the speed of sound at 0 degrees Centigrade and pressure of 1 atmosphere. The above coefficient is calculated in Equation 3.4 below.

High levels of humidity result into reduction of air density (water vapour is less dense than dry air) which that in turn results into higher speed of sound.

An example sound speed calculation based on formula 3.2 for dry air, pressure of 1 atmosphere and at 0^0 C results into [43]:

$$velocity\ of\ sound\ 0\ \mathrm{deg}\ 1\ Atm = \sqrt{\frac{1.4 * 1{,}012{,}930}{0.001293}} = 331.2m/sec \qquad \text{Eq 3.4}$$

At temperature of 20^0 C and pressure of 1 Atmosphere of dry air, the formula results into the known value of 343 m/sec.

Key points on acoustics theory: one general conclusion that can be reached from the theory on acoustics is that sound propagation is dependent on multiple parameters including pressure, temperature, humidity and the medium properties. Specifically in the case of aircraft acoustics where sound is generated from multiple noise sources [44], [45], it propagates to the environment in a manner that depends on altitude, air pressure, temperature, humidity, wind, interaction with the airframe and propagation via the airframe.

Although there are multiple noise sources onboard an aircraft [44], [45], one key source that affects many acoustic characteristics is the propulsion method used. The following section lists available propulsion methods with a view towards their impact on acoustics.

3.3 Propulsion types with a view towards acoustics

The underlying concept behind any aircraft propulsion system involves propelling a stream of gas towards the rear of the engine [46] which due to the law of conservation of momentum will in turn produce thrust and movement of the aircraft on the opposite direction i.e. forward. Note that the same method can also be applied in reverse with thrust reversers enabled to decelerate the aircraft after it has landed.

Each propulsion method has a different acoustic noise signature that will be analysed in turn later on in this chapter. The main propulsion types can be listed below [46]:

- Piston propeller engine
- Turbojet engine
- Turbofan Engine
- Turboprop Engine
- Turboshaft Engine
- Ramjet and Scramjet Engine
- Electric propulsion (batteries / fuel cells / electricity from a generator) and hybrid engines

Piston propeller engine: it is one of the first engine types widely used onboard an aircraft [46]. This is similar to the combustion engine widely used in motor vehicles converting the chemical energy of fuel via combustion into kinetic energy of pistons and rotation of the propeller as shown in Figure 3.1.

Turbo jet engine: this type of engine uses the air inlet in the front to maintain sufficient air flow [46] (see Figure 3.2). It compresses this incoming air using a compressor and feeds it into the combustion chamber where the temperature of the gas mixture increases to about 1000^0 C. The high temperature/pressure gas mixture is fed to a turbine maintaining rotation of the compressor and then expelled via the nozzle at the back producing thrust. This key function of the engine sucking compressing and expelling gas justifies its nickname as an “air breathing engine”.

Turbo fan engine: it is very similar to the turbo jet engine in structure and function (see Figure 3.3) with the main difference that a fan is placed in the front to suck additional air aiming to increase the efficiency [46]. Once again it is the fast flow of high temperature/pressure gas expelled from the nozzle that produces thrust.

Turbo propeller engine: it is similar in architecture to a conventional turbo jet engine (see Figure 3.4) with the main difference that it is not the gases exiting from the nozzle that produce thrust [46]. Instead these high velocity gases are used to rotate the propeller shaft via a turbine. A gearbox is needed to convert the high-speed low torque rotation of the turbine (typically 10,000 to 25,000 rpm) to the optimal lower speed high torque of the propeller (less than 2400 rpm). If the propeller was to spin supersonic it would cause excessive amount of noise and would be inefficient.

Turbo shaft engine: it is exceptionally similar in architecture and function to the turbo propeller engine with the main difference that turbo propellers are designed to support the weight and load of the propeller whereas the turboshaft engine drives a transmission shaft which is structurally supported by the vehicle [46] (see Figure 3.5). Most helicopters are using a turbo shaft engine to create lift.

Ramjets and scramjets: this is a relatively new technology that begins to be applied by upcoming aircrafts (NASA X-43, Lockheed SR-72). Ramjets and scramjets do not have a compressor to accelerate the air before combustion, instead they use the fast flow of air when the plane is already flying at high speed typically above Mach 1.0 [46] (see Figure 3.6). For this reason aircrafts like SR-72 use a conventional turbo jet engine to accelerate the aircraft and then switch to a separate ramjet/scramjet engine with all the engines sharing the same nozzle for the propulsion. Similarly when the aircraft decelerates the turbo jet engine takes over from the ramjet/scramjet engine.

Electric propulsion and hybrid engines: in alignment with increased levels of electrification of other transports, electric propulsion is gaining momentum either as form primary thrust generator or in the form of hybrid electric propulsion aside a conventional engine. Electric motors are therefore used to drive propellers using power from onboard batteries, fuel cells or generators.

Aircraft engine trends: Most commercial passenger aircrafts nowadays are using turbo fan engines due to their efficiency and relatively low acoustic noise. There is an upcoming trend (by Embraer, Bombardier and ATR) towards the increased use of turbo propeller engines as they can be more fuel efficient in short flights and lower speed [44]. One limitation of turbo props though is the high levels of acoustic noise. Research and development is being performed attempting to address this acoustic issue of turbo propellers. Piston propellers that are also considered noisy engines can still be found typically in smaller recreational and training planes such as Cessnas. However due to their relatively high number of moving parts, need for frequent service to remain reliable and their relatively lower power density versus jet engines they are not widely used for large commercial planes.

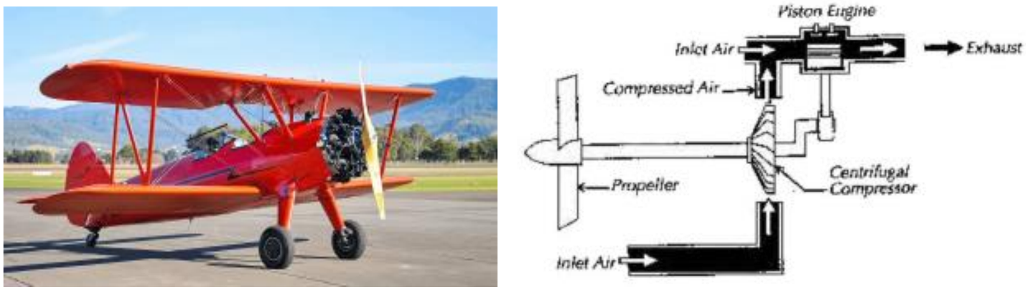


Figure 3.1 Piston propeller engine, credit [47], [48]

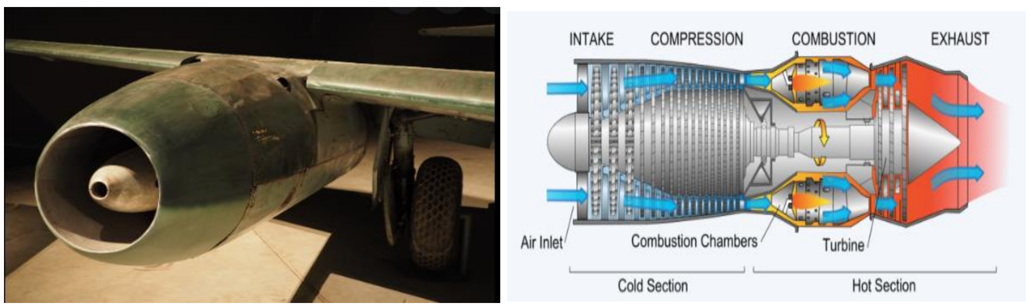


Figure 3.2 Turbo jet engine Junkers Jumo 004, credit [49], [50]

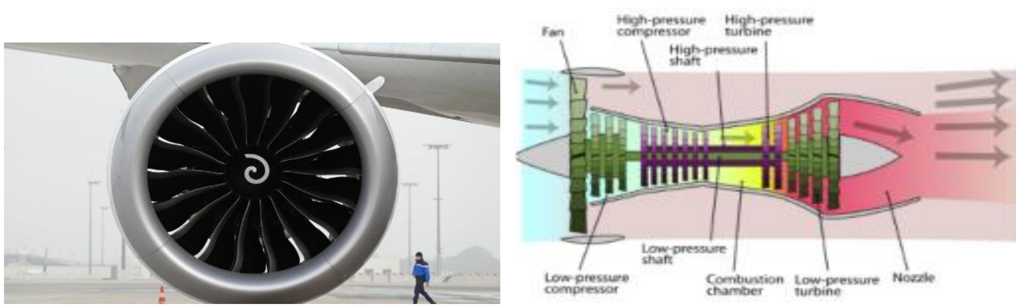


Figure 3.3 Turbo fan engine, credit [51], [52]

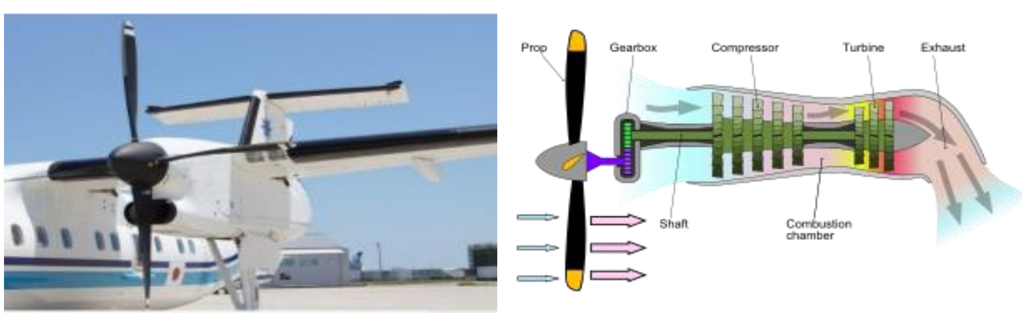


Figure 3.4 Turbo propeller aircraft engine, credit [53], [54]

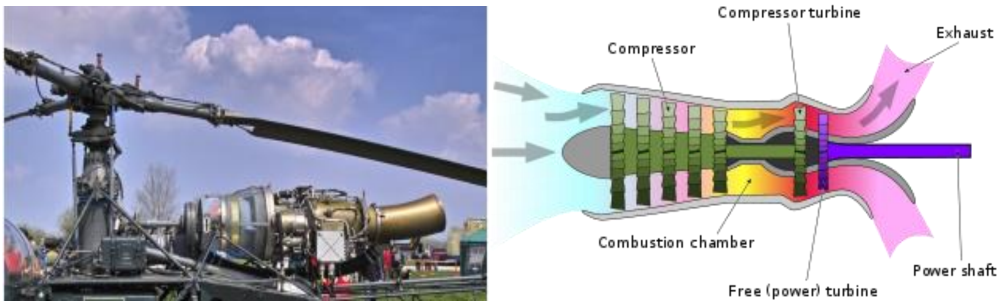


Figure 3.5 Turbo shaft engine onboard a helicopter, credit [55], [56]

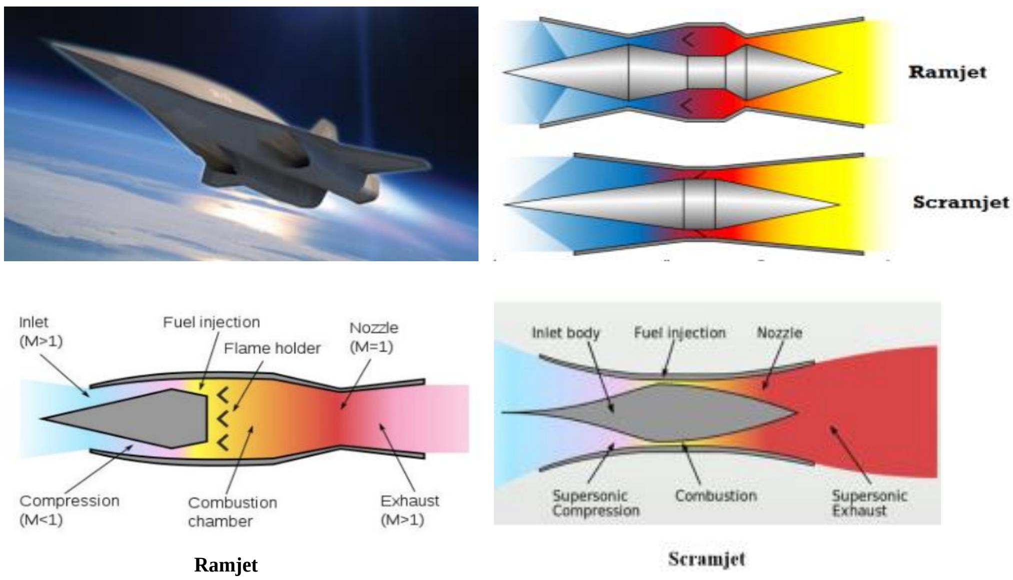


Figure 3.6 SR-72 under development combining turbojet, ram jet and scram jet, credit [57], [58], [59]

Aiming to understand aircraft acoustics, the above section listed typical aircraft propulsion methods. The next section details how the various components of each propulsion type contribute to the overall acoustic noise.

3.4 Aircraft acoustic noise sources

The acoustic noise originating from an aircraft can be divided into engine noise and airframe noise [44] , [45]. The engine noise can then be decomposed further to noise from the various internal parts of the engine contributing to the overall acoustics. All of the noise sources then propagate through the air and via the airframe.

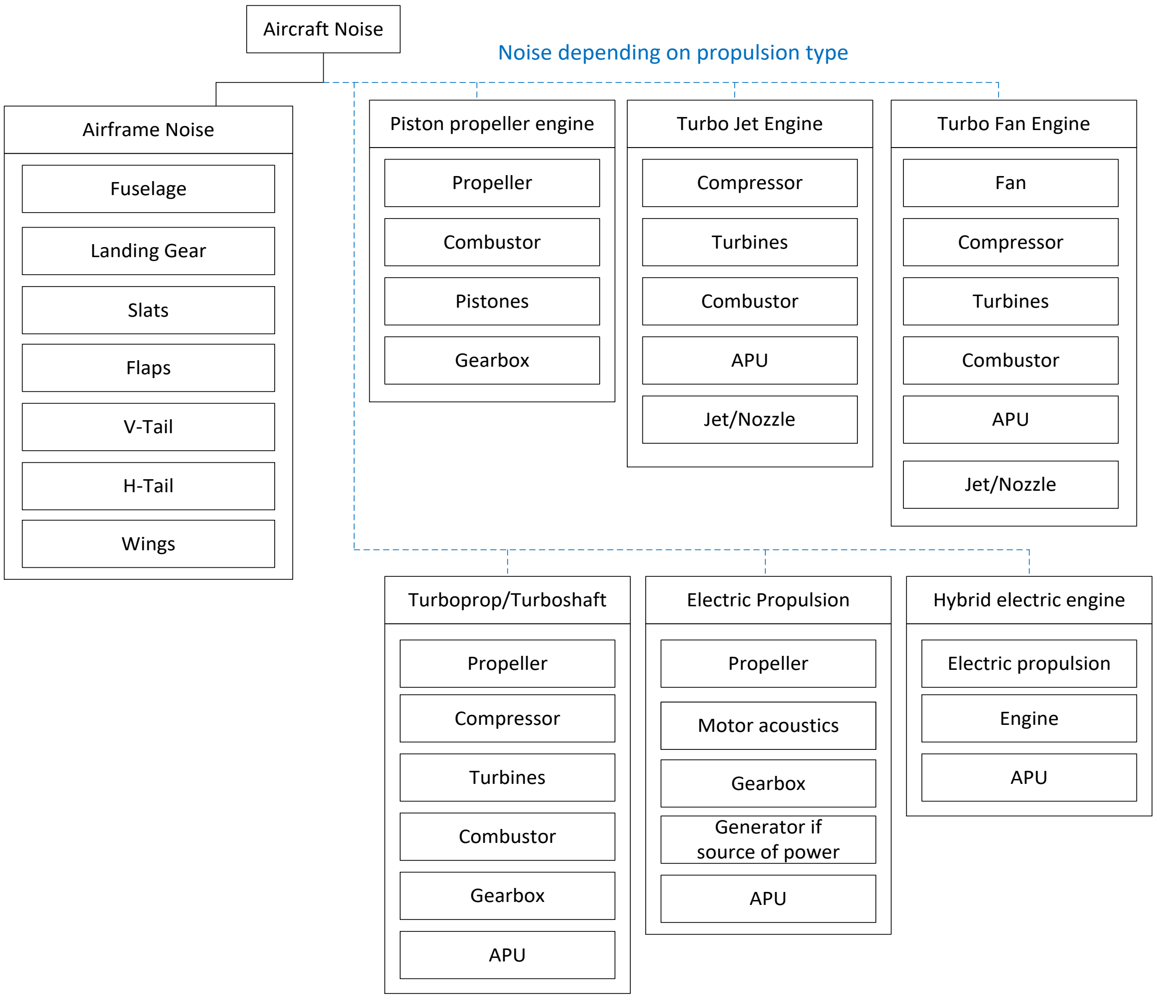


Figure 3.7 Acoustic noise sources

Figure 3.7 lists the various noise sources from an aircraft depending on the propulsion type used followed by a very brief characterisation of each noise type. This characterisation is useful to show applicability of the proposed acoustic improvement algorithms proposed in this thesis.

Compressor and turbine (turbo machinery) noise: two key components of the turbo-jet, turbo-fan, turbo-prop and turbo-shaft engine is the compressor and the turbine. These two components are also known as turbo machinery and are composed of rotor and stator (vanes) elements where the former rotate at close proximity to the latter [44]. The noise from these components is due to blade wakes and vortices that are generated during the rotation and the unbalanced pressure on the blades. This unbalanced pressure is in turn caused by disturbance of air inflow and turbulence [44], [45]. The vortices result into vibrations that propagate via the air and the nearby engine structure. The acoustic noise being generated has both tonal and broadband noise [50, 45]. The tonal noise is related to the rotational speed of the blades and broadband noise due to random conditions such as the turbulence resulting to unsteady airflow.

Combustion Noise: this noise originates from the combustion of fuel. The controlled explosion of fuel initiates sudden expansion of gas that creates air vibrations known as direct combustion noise [44]. This expansion in turn creates sudden variations of gas flow through other parts of the engine specifically the turbine and nozzle known as indirect combustion noise [44]. The noise from combustion is broadband covering a wide range of frequencies. This is due to the relative randomness for an external observer of this sudden air expansion.

Landing Gear noise: this is a significant noise source at approach to landing and take off for an aircraft [60]. The noise from a landing gear is first due to the motor enabling the gear and then due to vortex shedding created as the air flows to various components of the landing gear. The Landing Gear noise propagating through the air frame and air is mostly broadband due to relative randomness of the air flowing through various shaped components of the gear [60].

Jet Noise: this is one of the most dominant acoustic noise sources in aircrafts that do not have a propeller [44] , [45]. The noise at subsonic flight speed is due to the turbulent mixing of high-speed gas from the jet engine and low speed ambient air creating fluctuations in air pressure and noise perceived by nearby observers. At supersonic flight speed, shock noise is also observed where the high-speed air gas exiting from nozzles introduces powerful vibrations. Turbulent mixing jet noise observed in sub and supersonic flight speed is broadband in nature due to the randomness of the air disturbances. Shock noise at supersonic speeds contains both broadband and tonal components, tonal elements taking place due to resonant frequency and an acoustic feedback process taking place creating very loud high frequency noise.

Noise from aerodynamic surfaces: aerodynamic surfaces (aileron, rudder, elevator, slats, flaps, fuselage and wings) introduce acoustic noise as the plane flies and the air travels around them. Vibrations on these surfaces due to unsteady flow of air and air vortices initiate acoustic waves that can be heard both by the passengers and residents near airports. The mechanisms that control their position (motors or moving parts from hydraulics) can also introduce noise. The noise from the unsteady airflow on the surfaces is broadband in nature considering the random nature of turbulence and vortices. The noise from motors controlling them have tonal elements.

Propeller noise: this type of noise is of special interest as it covers a number of engine types (piston propellers, turbo-prop, electric/hybrid propulsion). It has both tonal and broadband elements [44], [45]. The tonal acoustic noise from a propeller is due to thickness noise, steady and unsteady loading noise. Thickness noise harmonic is due to the displacement of air molecules from the volume of propeller blades as they rotate. This displacement vibrates the air molecules and initiates the acoustic noise. Thickness noise is more evident at high rotation speed [44]. Other than general air displacement, the propeller blades are moving in an angle in the air creating thrust. This angular movement creates a moving pressure field of air travelling towards the rear of the aircraft. Steady loading harmonic noise is more observable at low to medium rotational speed [61]. Unsteady harmonic loading noise is caused by unsteady air pressure fields that is caused by turbulence, currents and wind [44]. Broadband propeller noise is mostly caused by turbulence [44]. Tonal propeller acoustic noise is more dominant than broadband propeller noise.

Key points on aircraft acoustics and noise sources: acoustic noise from an aircraft does not have one single source but is a combination of multiple sources. These sources are very dependent on the propulsion type and this is the reason why the noise from a commercial turbo-fan jet engine is so different from a recreational piston engine propeller aircraft. Each of the aircraft reflects the technological advancements at the time of their introduction, efficiency demands, reliability/service needs, acoustic considerations etc. The acoustic noise generated from an aircraft is of both tonal and broadband frequency spectrum.

Addressing acoustic noise is a key problem in aerospace. The following section of this chapter introduces two proposed acoustic improvement algorithms namely:

- a variant of the Pseudo Random High Frequency Injection (PRHFI) algorithm and

- Active Noise Cancellation (ANC) by means of High Frequency Injection (HFI).

The former algorithm attempts to advance the current state of the art in the area while the second algorithm is a novel method conceived, named and investigated in this research effort.

The details of these algorithms along with simulation and test results are provided in chapters 7 and 8 of this thesis respectively.

3.5 Improving acoustics I, the proposed PRHFI variant

High Frequency Injection (HFI) is a sensorless method that utilises the saliency of a motor to identify the rotor angle. The method involves superimposing a high frequency voltage component to the fundamental frequency of the motor's phase voltages. This injection can take place in the stationary (Figure 3.8) or synchronous frames (Figure 3.9). The injection signals can also vary in shape specifically sinusoidal and square.

Stationary frame sinusoidal injection formula:

$$\hat{V}\alpha_{injected} = V_{carrier} * (sin\omega_c t) \qquad \text{Eq 3.5}$$

$$\hat{V}\beta_{injected} = V_{carrier} * (cos\omega_c t) \qquad \text{Eq 3.6}$$

Synchronous frame sinusoidal injection:

$$\hat{V}d_{injected} = V_{carrier} * (sin\omega_c t) \qquad \text{Eq 3.7}$$

Note: Refer to Appendix C of this thesis for the transform equations between stationary and synchronous frames.

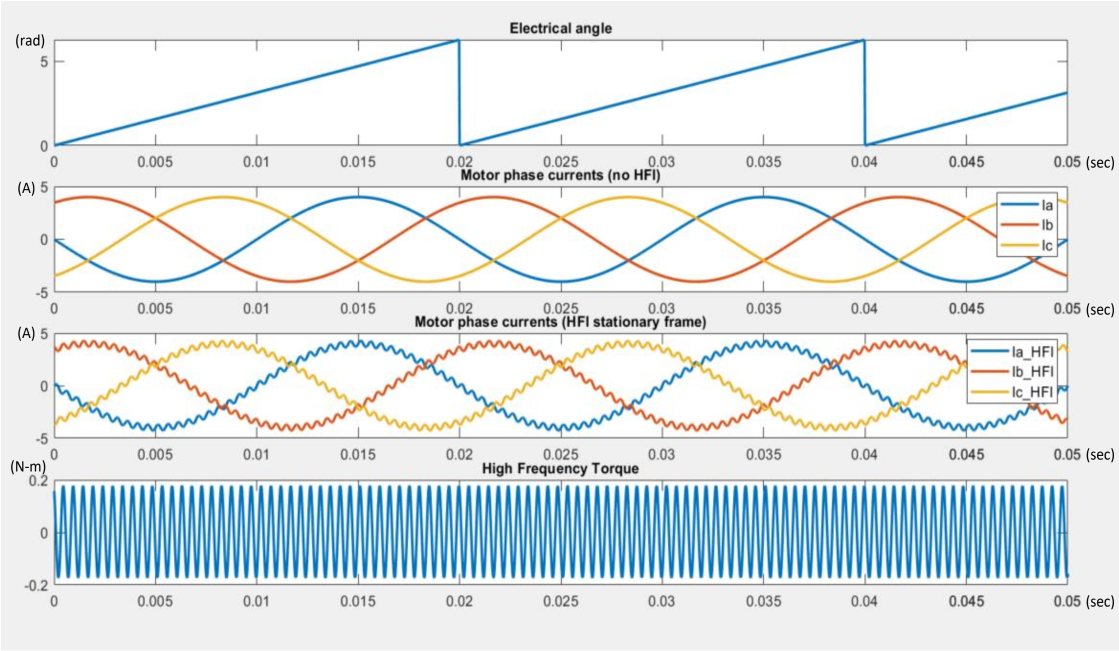


Figure 3.8 High Frequency Injection stationary frame based, phase currents and torque

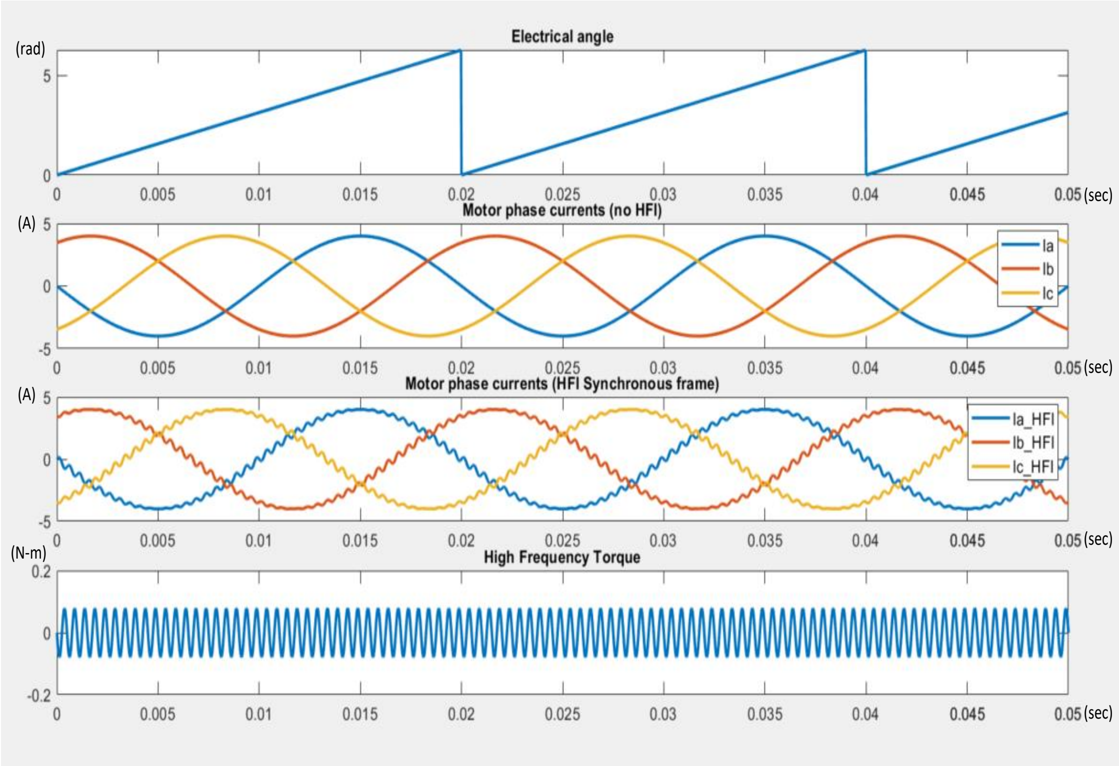


Figure 3.9 High Frequency Injection synchronous frame and motor phase currents

Note that two Matlab models were written to illustrate the stationary and synchronous frame HFI shown in figures 3.8 and 3.9, please refer to Appendix B.1.3 and B.1.4 of this thesis to obtain the Matlab code.

Note1: Looking at figure 3.8 it can be noticed that the injection amplitude in the stationary frame results into constant injection amplitude to the three phases as the motor is spinning. This type of injection also results into higher levels of torque pulsation.

Note2: Looking at figure 3.9 the HF current injection amplitude in the synchronous frame results into varying injection amplitude to the three phases as the motor is spinning as the transfer function from synchronous to stationary frame (Appendix C) is a function of the rotor angle. This type of injection also results into lower levels of torque a characteristic mentioned in section 2.2.2.2.

Note3: Slightly different variants exist in in literature of the HFI formula in the synchronous frame (Eq 3.7). For example [62] refers to injection term of $-sin\omega_c t$, [63] refers to injection of $sin\omega_c t$ and [64] refers to $cos\,\omega_c t$, however the fundamental underlying principle holds in all three methods with some minor differences in the mathematical equations of the demodulation steps.

Torque pulsation and acoustics: Both stationary and to a lesser extent synchronous frame HFI result into high frequency currents and high frequency torque pulsations:

$$T_{PMSM} = (3/2) * (P/2) \ * (\lambda_m * I_q + \left(L_d - L_q\right) * i_d * i_q) \qquad \text{Eq 3.8}$$

where P is the number of poles, λ_m is the permanent magnet flux, L_d, L_q is the d and q axis motor inductance and i_d, i_q is the d and q axis motor current.

These additional torque pulsations in turn generate acoustic noise that is audible to the human ear. Research has shown that the human ear is more susceptible to pure tones of a specified audible frequency rather than random noise of equivalent amplitude [65]. An area of research in the past years has thus been to vary the injection frequency in a pseudo random manner and therefore injecting a spectrum of frequencies [62], [63] resulting into the torque pulsations and acoustic noise spreading into a wider audio spectrum. A variant of the Pseudo Random High Frequency Injection (PRHFI) algorithm is therefore presented in chapter 7 of this thesis to reduce perception of HFI acoustic noise.

3.6 Improving acoustics II, Active Noise Cancellation by means of HFI

Acoustic noise has been an area of interest in a wide range of industries. Active Noise Cancellation or Control (ANC) is a method of injecting acoustic noise that is in anti-phase relation and cancels the primary acoustic noise residing in a system. The method has been applied from low-cost Noise Cancellation earphones to state of the art ANC systems on board aircrafts. Active Noise Cancellation typically involves installation of microphones to capture the noise that resides within a system and speakers to generate noise that is in antiphase relation. A novel method is proposed in this thesis where the acoustic noise from HFI is used to cancel existing noise within a system. The proposed HFI ANC process can address both broadband and tonal noise but it was tested in this research for tonal audio signals only. Tonal noise as explained in previous section is dominant in electric/hybrid propulsion so it considered an experiment of interest and broadband HFI ANC was included in the Future Work section of this thesis. The HFI ANC algorithm along with test results on the HEMAS platform are presented in chapter 8 of this thesis.

3.7 Conclusions

This chapter provided a background on aircraft acoustics which is one of the key areas investigated in this research effort. An analysis of aircraft acoustics in the form of decomposition to its sources has been presented aiming to analyse typical acoustic noise that is generated from an aircraft. Aiming to suppress acoustic noise, two algorithms were introduced namely the Pseudo Random High Frequency Injection (PRHFI) and Active Noise Cancellation (ANC) by means of High Frequency Injection (HFI). Each of these algorithms will be detailed and presented along with experimental results in chapters 7 and 8 of this thesis.

Chapter 4 Experimental Method

4.1 Introduction

This chapter aims to provide an insight to the platform used for hardware testing as well as the test methodology that was followed. The HEMAS system and test rig are therefore presented in the first sections. The software design environment is then introduced with a particular focus on features that were used during testing. The test set up and methodology are finally presented to the reader giving an insight into the test steps during experimentation.

4.2 The experimental system

4.2.1 Helicopter swash plate mechanism

The rotor of a helicopter and specifically the rotation of the rotor blades is used to establish the necessary lift for the helicopter to fly. In order to control the direction, altitude and speed of flying, the angle of the rotor blades needs to be adjustable. The pilot is able to control the angle of the blades using a device known as swashplate. The swashplate is situated below the rotor blades as per Figure 4.1 and can change the angle of the blades individually or collectively as they revolve. This allows the helicopter to move in any direction around a 360-degree circle, move forward, backward, left right or change of altitude.

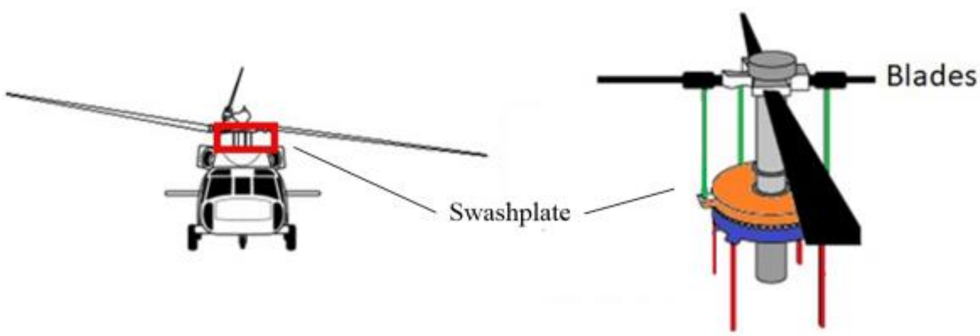


Figure 4.1 Helicopter swash plate (Credit spinningwing.com)

The swashplate belongs to a group of devices known as primary controls as it is an essential system used to control the flight of the helicopter. Primary control systems including the swashplate have been in the past decades controlled using hydraulics. Aiming to reduce the weight, improve fuel consumption and in alignment with a general trend of More Electric Aircraft (MEA), the HEMAS actuator platform system was developed controlling the swashplate with the aid of electric motors. The HEMAS system described in the following section is the hardware that was used as a test platform to experiment with reliability and acoustic enhancement methods proposed in this research. Note that the HEMAS hardware platform was not developed under this research project but the DSP code was extended and hardware experimentation on the platform took place aiming to investigate and evaluate the methods proposed in this research.

4.2.2 The HEMAS hardware system and test rig

The Helicopter Electro-Mechanical Actuation System (HEMAS) is used to control the swash plate of a helicopter [18], [19], [20]. The HEMAS platform was not developed under this research and was not developed by the author of this thesis. It was designed as part of the European Clean Sky JTI Research Programme and in collaboration between EADS-IW, Eurocopter, Liebherr-Aerospace and AgustaWestland [66].

The complete HEMAS system is composed of three actuators placed circumferentially offset to each other by 120^0 [67]. Each actuator is formed by two motors M1 and M2 (see Figure 4.1) and each motor's shaft is connected to a gearbox and a ball screw converting rotary to linear movement [18].

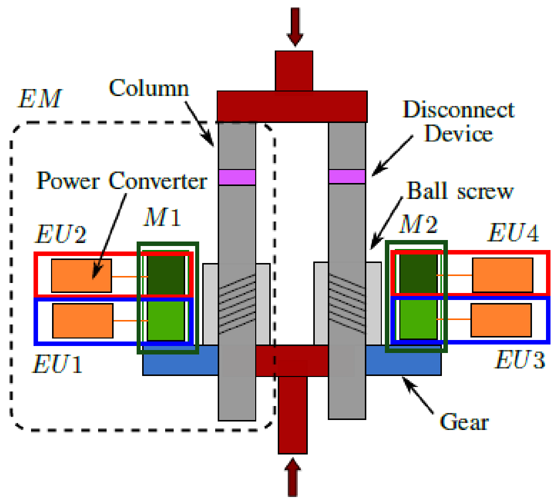


Figure 4.2 HEMAS actuator two motor system

The HEMAS actuator was designed to accommodate demanding constraints including high power density, high performance, fault tolerance and high temperature of operation. The system's target application is within the hottest region of the helicopter ($T_{ambient}$ from -30^0C to 110^0C) [68]. With respect to the high power density constraint, the motor designed for the system is an SPMSM, whose structure can naturally support such high levels of power density [67], [18]. With respect to fault tolerance, the motor has a double three phase winding arrangement to allow safe operation in case of a short circuit to the windings [18]. With respect to high temperature, liquid cooling was not used so as not to compromise reliability and increase mass. Instead, the motor is using natural air convention as the cooling method, suitable materials and appropriate design techniques aiming to accommodate the required temperature range [18].

For test purposes of the motor and control, a test rig was developed that incorporates only one motor rather than the full 3 actuator x 2 Motors. It is this rig that was used for the testing in this research, see Figure 4.3. Note that the HEMAS control electronics can be seen in Figure 4.4.

The parameters of the HEMAS SPMSM are described in Table 4.1. The motor saturation saliency characteristics used in Chapter 6 were obtained by importing the motor design in the Motor-CAD software.

The spinning of the HEMAS motor is controlled by analog and digital hardware cards shown in Figure 4.4. The Power Electronics card holds mostly analog and high-power electronics such as the converter with the IGBTs. The Control Circuit board holds mainly digital/mixed signal domain ICs such as ADCs, an FPGA, and a DSP.
Note that the scaling for the ADCs and resolver demodulation designs above are:
- 22mV per Least Significant Bit (LSB) for the 270V DC Link interface.
- 7.8mA per LSB for the motor phase currents.
- 0.08789 degrees mechanical and 0.43945 degrees electrical per LSB.
- 0.85 degrees Centigrade per LSB.

The sampling noise levels of the digitization process were found to be in the region of 2-3 Least Significant Bits for each interface.

4.2.3 Software environment and outline of the controller

The software motor controller is based on the Infineon TriCore 1797 Digital Signal Processor (DSP) and development board. The DSP controller is a centralized part in the system in a similar way as is the brain for a human or animal. Following the processing of data originating from external sensors, the digital controller enables the switching of IGBTs controlling motor phase currents, torque and rotation of the motor.

The main functional steps of the software controller ensuring stable control of the HEMAS SPMSM are shown below and they align with conventional speed and current control loops of PMSM control:

- capture the motor's electrical angle from the resolver ADC and calculate speed feedback
- receive speed demand from Comms interface. Calculate speed error by subtracting speed demand – speed feedback. Calculate *Iq_demand* by feeding the speed error to a PI controller
- capture the motor phase currents from ADCs *Ia*, *Ib*, *Ic*. Use Clarke and Park transform, to convert the motor phase currents to *Iq_feedback*, *Id_feedback*
- feed the Id/Iq current error to a PI controller calculating *Vd/q_demand*
- apply inverse Clarke/Park transform to convert *Vq_demand*/*Vd_demand* to *Va_demand*, *Vb_demand*, *Vc_demand* used to drive the 3 phase stator motor supply lines using Space Vector Modulation (SVM)
- repeat the steps above.

Note Appendix C provides information and equations for Clarke/Park transforms and Appendix D provides information for the Space Vector Modulation step

Parameter	Value
No. of pole pairs p	5
Maximum speed	5200 [rpm]
Rated Current	4.7 [A]
Peak Current	34 [A]
Peak Power	2.6 [kW]
Efficiency	98.8 [%]
Phase Resistance R_S	0.23 [Ω]
Phase Inductance L	1.193 [mH]
Voltage constant κ_E	0.092 [V/rads]
Torque constant κ_T	0.142 [Nm/A]

Table 4.1. HEMAS motor parameters

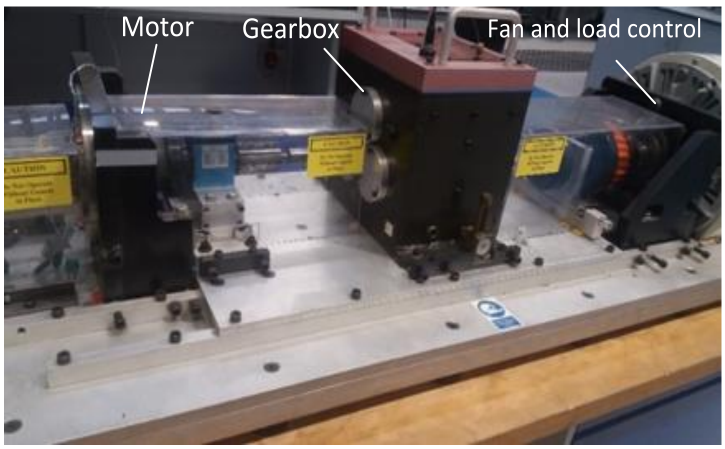


Figure 4.3 HEMAS test rig

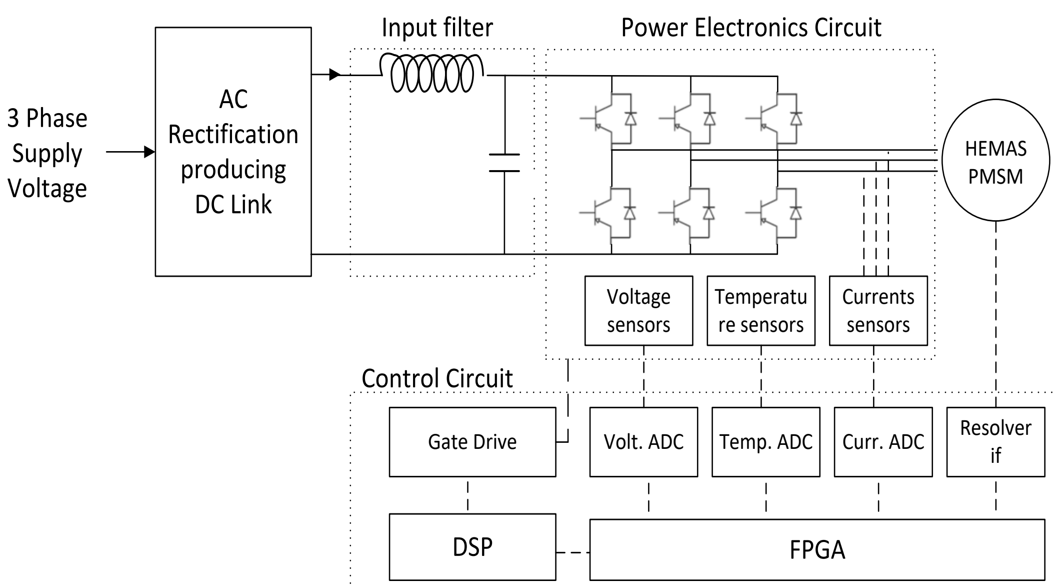


Figure 4.4 HEMAS system diagram and circuits

The software environment that was used to implement and debug the digital design is the Eclipse for TriCore software tool. The HEMAS controller design was developed in the C language with small number of machine level instructions to accommodate hardware constructs of the DSP.

Note that the design does not contain a position control loop outside the speed and current control loop. The function of the system is to control the motor to the demanded speed.

The primary means of debugging the software design within the Eclipse platform, is watch windows where C variables can be monitored and edited at run/time by the user. The variables added in watch windows during this research can be generally grouped into two types, Read/Write (RW) and Read Only (RO) variables as shown in Figure 4.6. The former group intends to increase controllability during run-time, as it allows the user to enable/disable execution of statements using these variables. The latter group aims to increase observability and allows the user to capture the state of various internal variables.

4.3 Data acquisition

4.3.1 Controller data acquisition

To analyse the controller functionality and evaluate the proposed sensorless and audio algorithms, it is of great significance to be able to monitor the internal variables of the DSP controller. The code was therefore extended with an array of RO registers `regN[no_of_samples]` that are used to capture samples of internal variables of interest such as phase currents, resolver angle/speed, sensorless observer angle/speed and HFI signals.

This sampling/capturing is performed at 10kHz and is initiated when the user updates `capture_data` variable from 0 to 1. The array of RO variables `regN[no_of_samples]` that stores the samples is then saved in a csv file that that is later processed in Excel and Matlab.

Variables of interest for sensorless and HFI data that were captured during experiments include:

- Resolver speed
- Sensorless observer speed
- Resolver angle
- Sensorless observer angle
- HFI signals

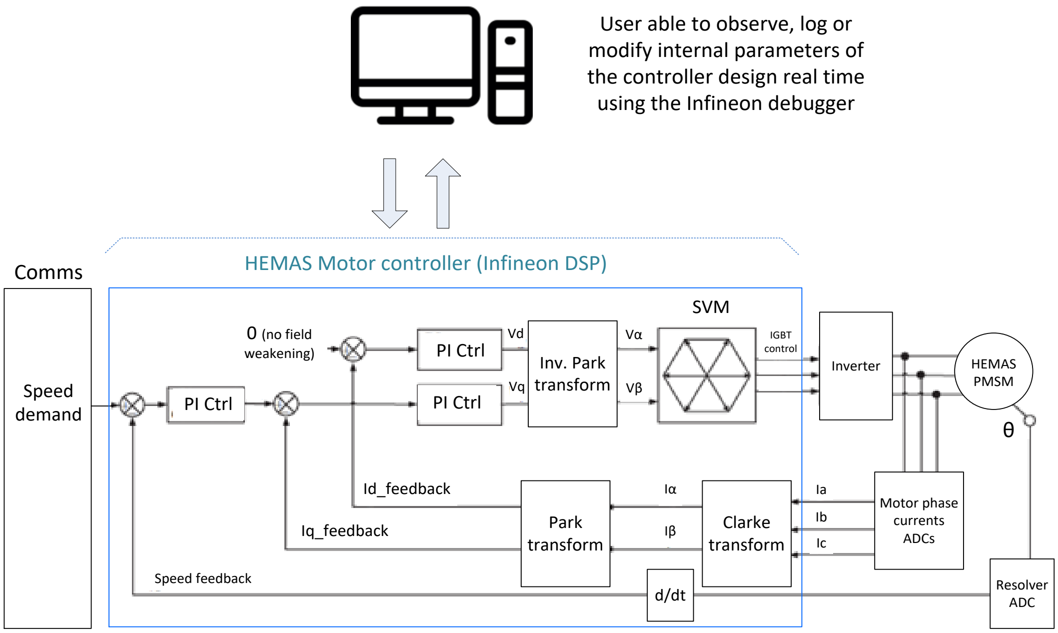


Figure 4.5 Block diagram of the HEMAS controller topology

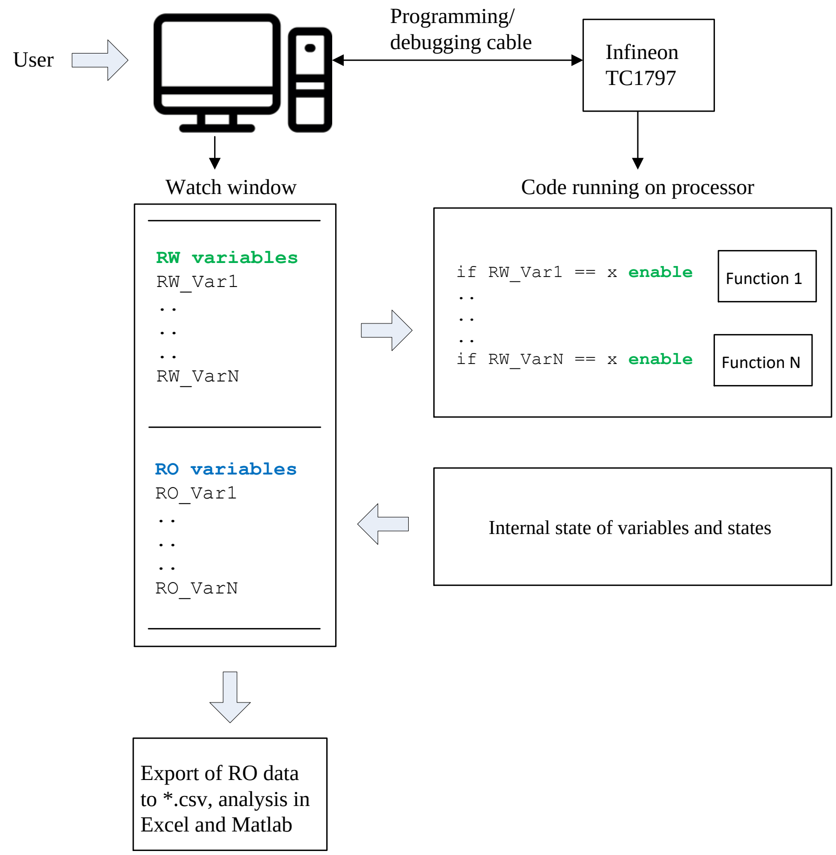


Figure 4.6 Usage of watch windows for controllability and testability

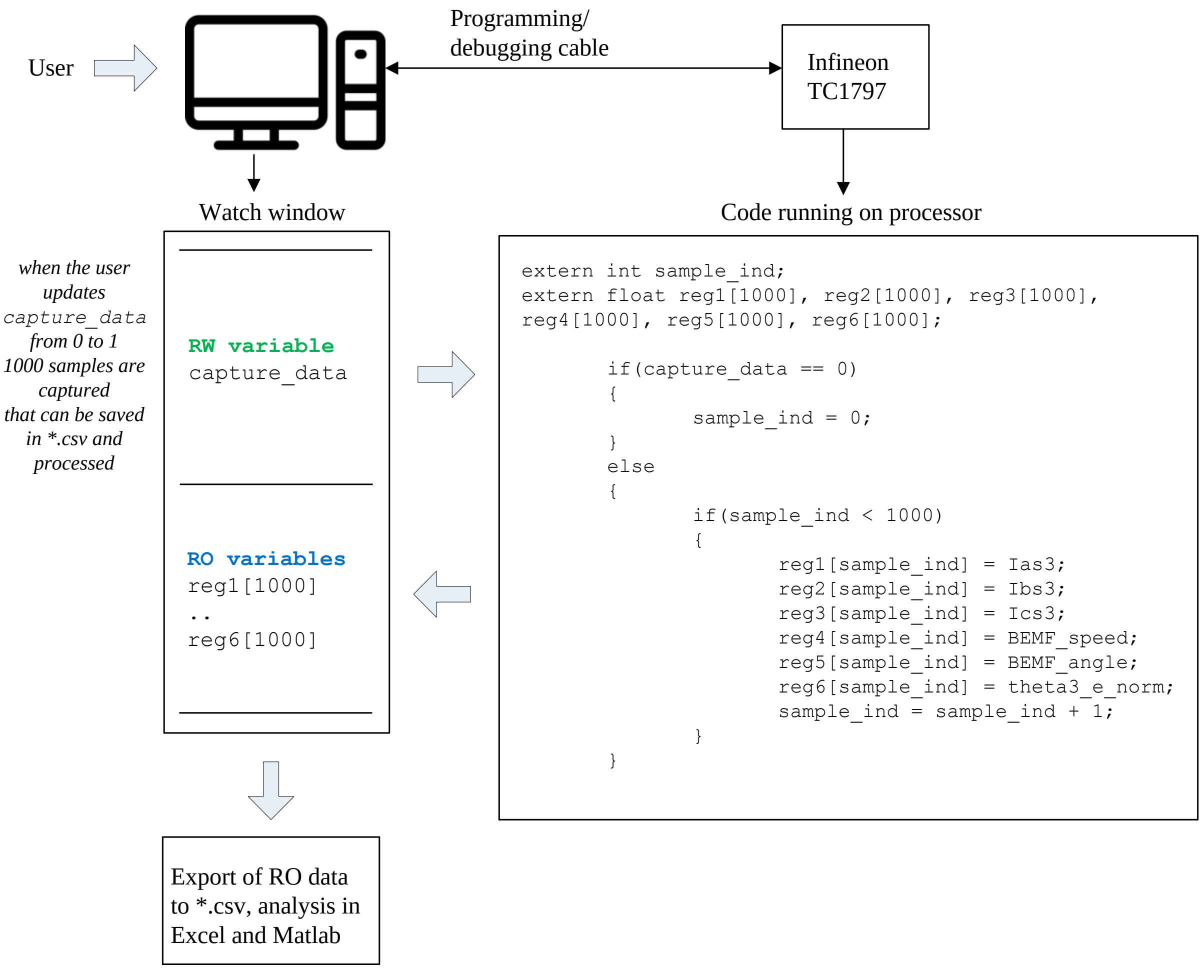


Figure 4.7 Usage of watch windows for sensorless observer testing

4.3.2 Acoustic audio capturing and acquisition

A key area investigated in this research is algorithms that can reduce the acoustic noise or its perception within a motor drive. To evaluate the effectiveness of the proposed algorithms, an audio capturing and data acquisition method is needed.

To minimize cables that can be susceptible to EMI from gate drive switching and taking also into account the advancement in consumer electronics, audio is captured using a general purpose microphone within a mobile phone. To evaluate the audio capturing method, Appendix E of this thesis provides details on the hardware used for audio capturing and a set of tests illustrating how the microphone responds to audio generated in Matlab. The acoustic capturing method involves sampling audio at 48kHz.

After testing, the digital audio is imported in Matab and processed.

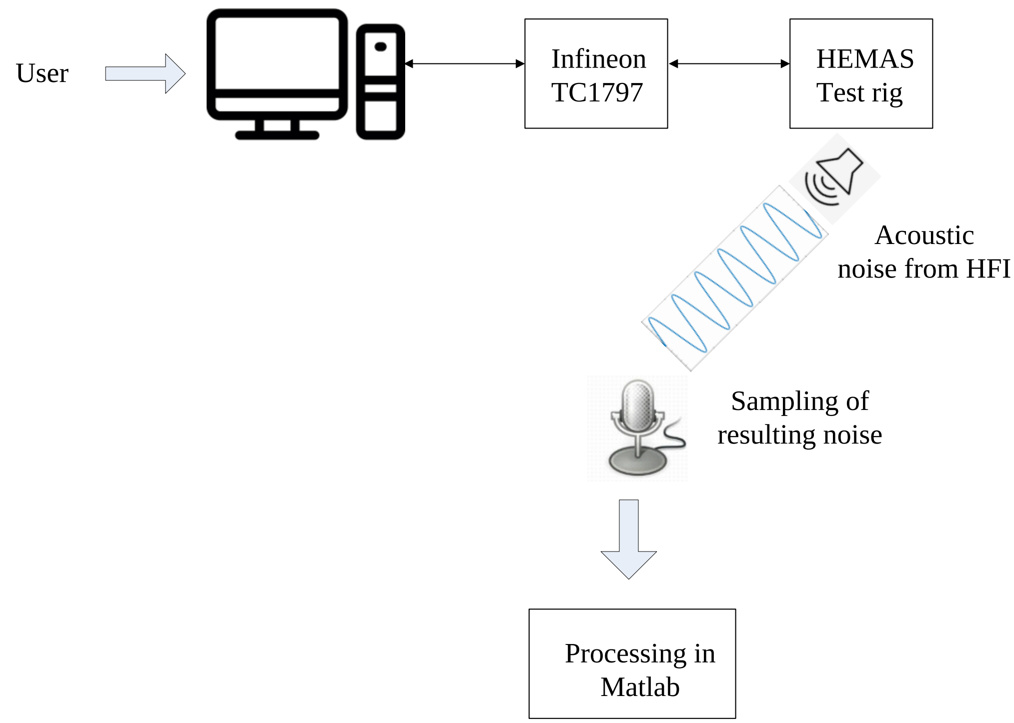


Figure 4.8 Audio data capturing, data acquisition and processing

4.4 Conclusions

The chapter first introduced the HEMAS platform that was used for hardware testing in this research. It then provided a summary of its key hardware and software elements. The test set up was then outlined regarding data acquisition from the controller and audio sampling using a microphone. The DSP data and audio acquisition method along with its processing in Matlab are important steps towards evaluating the reliability and acoustic improvement methods proposed in this research.

Chapter 5 Model Based Observer

5.1 Introduction

This chapter describes the first sensorless algorithm proposed in this thesis towards enhancing system availability. The proposed method is a model based observer and therefore can function when the motor is spinning at medium to high speed. The chapter first provides a theoretical background of the method detailing the PMSM's equivalent circuit and related mathematical formulas. It then continues by identifying where this method is situated within the overall proposed hybrid sensorless observer. VHDL Modelsim simulation results of the algorithm are then presented targeting the HEMAS platform motor parameters. The implementation steps of the observer in C are subsequently described illustrating how the algorithm was mapped on the targeted DSP platform. Hardware testing results of the sensorless method on the HEMAS test rig are finally presented analysed and compared with simulation artefacts.

5.2 Theoretical background on motional Back Electro Motive Force

The controller of a PMSM is designed to switch on/off the inverter switching devices (IGBTs/Silicon Carbides) creating motor phase currents and synchronizing stator and rotor magnetic fields producing torque as per equation:

$$T_{PMSM} = (3/2) * (P/2) * (\lambda_m * I_q + (L_d - L_q) * i_d * i_q) \quad \text{Eq 5.1}$$

where P is the number of poles, λ_m is the permanent magnet flux, L_d, L_q is the d and q axis motor inductance and i_d, i_q is the d and q axis motor current.

While the controller regulates the voltage application to each of the motor's phases to control the currents and torque, this supplied voltage is dissipated in three voltage drops [8]:

- Voltage drop related to Ohm's law and due to stator's resistance *Rs*.
- Voltage drop due to the stator's inductance response to *Ls * dI / dt*.
- A voltage drop known as Back Electro Motive Force (EMF) Voltage.

The electrical circuit associated with this series of voltage drops is also known as the motor's equivalent circuit and can be expressed in either the stationary or synchronous frames. The equivalent circuit and associated mathematical formulas can be expressed in the 3-phase stationary frame (a, b, c), in the 2-phase stationary (alpha, beta) and in the synchronous frame (d, q axis) as shown in Equations 5.2, 5.3, 5.4 respectively.

$$\begin{bmatrix} V_a \\ V_b \\ V_c \end{bmatrix} = R_s * \begin{bmatrix} i_a \\ i_b \\ i_c \end{bmatrix} + L_s * \frac{d}{dt}\begin{bmatrix} i_a \\ i_b \\ i_c \end{bmatrix} + \begin{bmatrix} E_a \\ E_b \\ E_c \end{bmatrix} \quad \text{Eq 5.2}$$

where R_s, is the stator resistance, L_s is the stator inductance, i_a is the motor's phase A current, i_b is the motor's phase B current, i_c is the motor's phase C current, E_a is the motor's back EMF for phase A, E_b is the motor's back EMF for phase B, E_c is the motor's back EMF for phase C.

$$\begin{bmatrix} V_\alpha \\ V_\beta \end{bmatrix} = R_s * \begin{bmatrix} i_\alpha \\ i_\beta \end{bmatrix} + L_s * \frac{d}{dt}\begin{bmatrix} i_\alpha \\ i_\beta \end{bmatrix} + \begin{bmatrix} E_\alpha \\ E_\beta \end{bmatrix} \qquad \text{Eq 5.3}$$

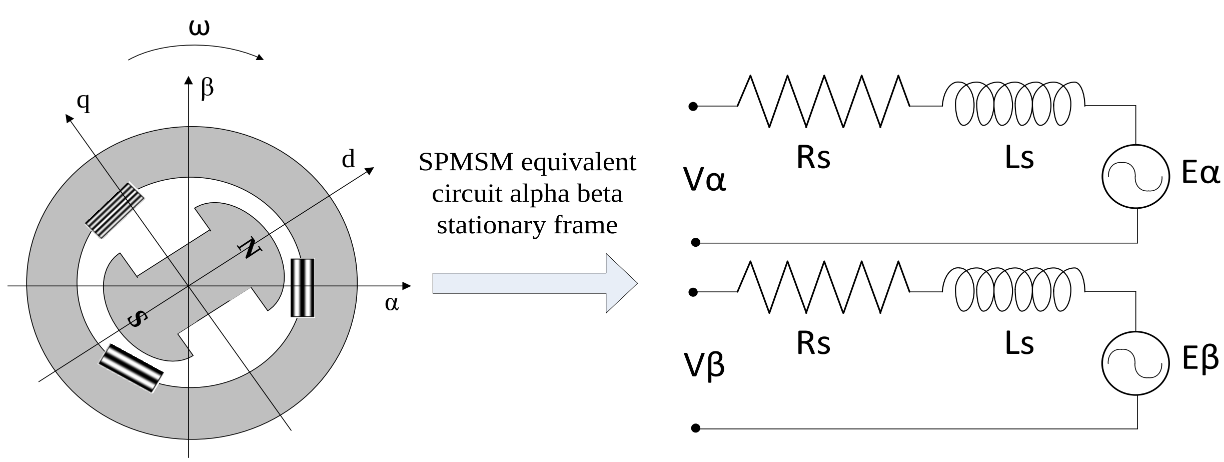
where R_s, is the stator resistance, L_s is the stator inductance, i_α is the motor's current in the alpha stationary frame, i_β is the motor's current in the beta stationary frame, E_α is the motor's back EMF in the alpha stationary frame, E_β is the motor's back EMF in the beta stationary frame.

$$\begin{bmatrix} V_d \\ V_q \end{bmatrix} = R_s * \begin{bmatrix} i_d \\ i_q \end{bmatrix} + \begin{bmatrix} L_d & 0 \\ 0 & L_q \end{bmatrix} * \frac{d}{dt}\begin{bmatrix} i_d \\ i_q \end{bmatrix} + \begin{bmatrix} 0 & -\omega * L_q \\ \omega * L_d & 0 \end{bmatrix} * \begin{bmatrix} i_d \\ i_q \end{bmatrix} + \begin{bmatrix} 0 \\ \omega * \lambda_m \end{bmatrix} \qquad \text{Eq 5.4}$$

where R_s, is stator resistance, L_d is the d-axis inductance, L_q is the q-axis inductance, i_d is the motor's current in the d-axis synchronous frame, i_q is the motor's current in the q-axis synchronous frame, λ_m is the permanent magnet flux and ω is the electrical frequency of the motor.

Depending on the modelling needs and known system parameters, one of these equivalent circuits is often selected. For example, if the rotor angle is unknown, the motor phase voltages and currents are known and $L_d = L_q$, the stationary frame equivalent circuit provides an equation where most parameters are known values. When $L_d \neq L_q$ or the focus of the modelling process is on Torque, the synchronous frame is chosen instead.

It is possible to switch between the 2/3 phase stationary and synchronous frames converting the voltages and currents using the Forward/Inverse Clarke and Park Transforms listed in the Appendix C of this thesis.



HEMAS Parameter	***Value***
Phase Resistance R_S	*0.23 [Ω]*
Phase Inductance Ls	*1.193 [mH]*

Figure 5.1 SPMSM equivalent circuit in alpha/beta stationary frame

Note that the motor parameters and saturation-based saliency table as will be detailed in section 6.4.2 were identified by importing the motor design into Motor-Cad. Also L_S = L_d = L_q.

The focus of this chapter is to present a sensorless method that estimates the rotor angle of a non-salient SPMSM and is therefore focusing on the equivalent model of the motor in the stationary frame as shown in Figure 5.1. The 2 phase (alpha, beta) frame is chosen versus the 3-phase as the relation of alpha versus beta is used to calculate the Back EMF and rotor angle using an atan function.

For a motor with currents i_α, i_β, stator inductance L_s, stator Resistance R_s, back EMF voltages $E\alpha$, $E\beta$ supply voltages V_α, V_β, the voltage drop distribution is shown in equation 5.3. The equation shows mathematically that the summation of a motor's back EMF voltage, voltage drop from resistance and voltage drop due to stator inductance response results into the supply voltage at the frame.

The amplitude of the BEMF voltage in a PMSM $E\alpha, E\beta$ is proportional to its rotational speed and a function of the rotor angle as shown in equation below:

$$\begin{bmatrix} E\alpha \\ E\beta \end{bmatrix} = k_E \omega * \begin{bmatrix} -\sin(\theta) \\ \cos(\vartheta) \end{bmatrix} \qquad \text{Eq 5.5}$$

where k_E is the Back EMF constant, ω is the electrical frequency of the rotor and θ is the electrical angle of the rotor.

The BEMF voltage is a sinusoid (see Note1 below) whose amplitude is proportional to the motor's speed and a function of the rotor angle. Figure 5.2 illustrates this relation between the motor's speed, angle and back EMF voltage for the HEMAS motor taking under consideration the motor parameters shown in table 5.1.

Combining equations 5.3 and 5.5 results into Eq 5.6.

$$\begin{bmatrix} V_\alpha \\ V_\beta \end{bmatrix} = R_s * \begin{bmatrix} i_\alpha \\ i_\beta \end{bmatrix} + L_s * \frac{d}{dt} \begin{bmatrix} i_\alpha \\ i_\beta \end{bmatrix} + k_E \omega * \begin{bmatrix} -\sin(\theta) \\ \cos(\vartheta) \end{bmatrix} \qquad \text{Eq 5.6}$$

A matlab model was developed with the HEMAS motor parameters to illustrate how the electrical angle and mechanical speed relate to the back EMF voltage. The Matlab simulation is located in the appendix section and the simulation result is shown in Figure 5.2.

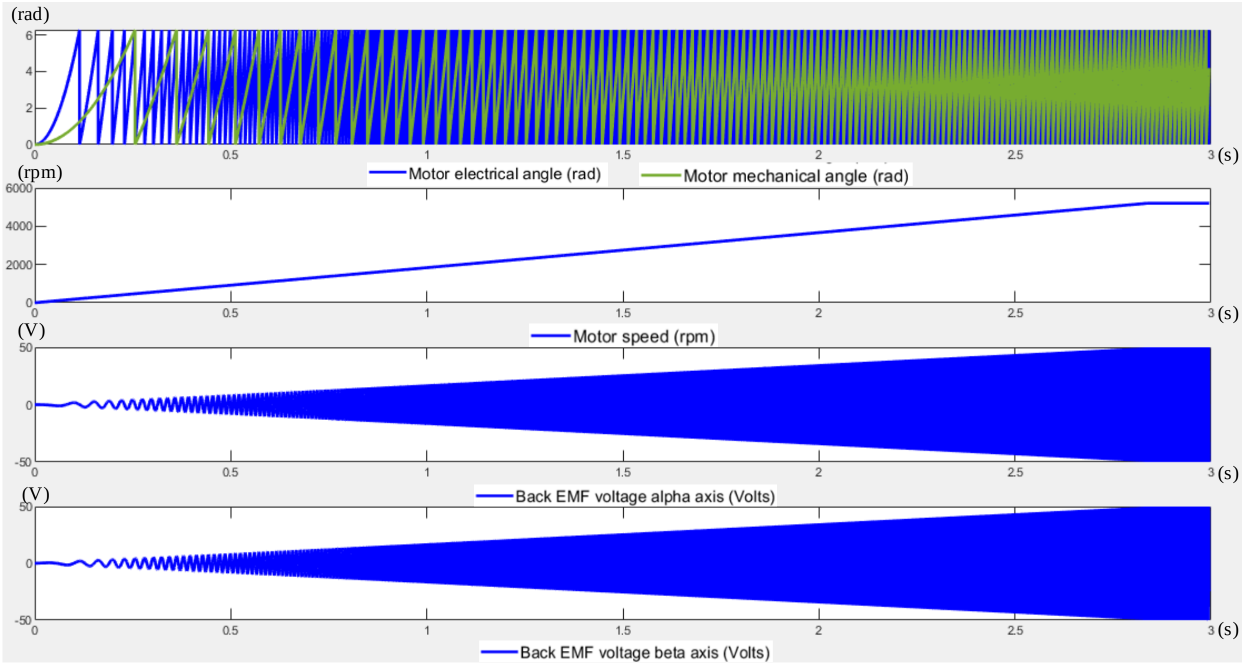


Figure 5.2 Back EMF voltage versus angle and speed for the HEMAS motor

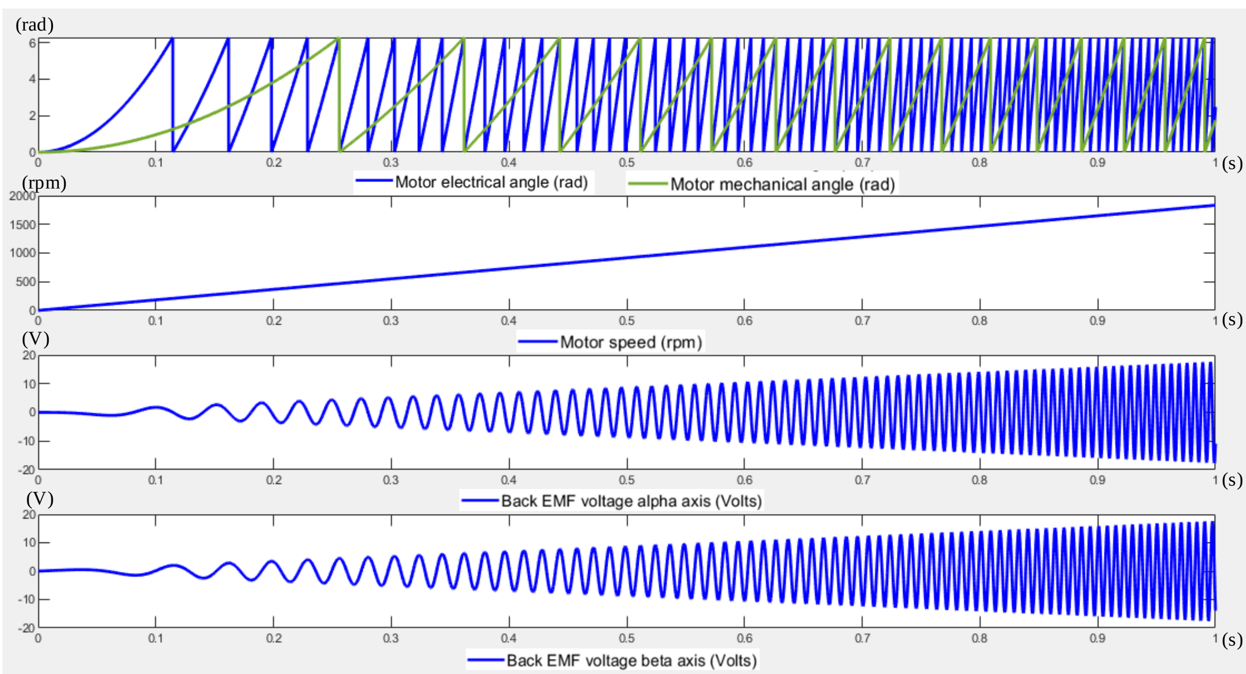


Figure 5.3 Back EMF voltage versus angle and speed for the HEMAS motor (zoomed in)

HEMAS parameter of interest used in simulation	***Value***
No. of pole pairs p	*5*
Maximum speed	*5200 [rpm]*
Voltage constant K_E	*0.092 [V/rads]*

Table 5.1. HEMAS motor parameters (short list)

*Note that a Matlab model *.m file was written to illustrate the relationship between motor speed, rotor angle and back EMF voltage for the HEMAS motor in figures 5.2, 5.3. Please refer to Appendix B1.2 of this thesis to obtain the Matlab source code.*

Note1: while motors are theoretically expected to have sinusoidal back EMF voltage, in practice they manifest back EMF harmonics that are superimposed to the fundamental sinusoidal BEMF and often distort its shape as observed from the controller.

Note2: While the above equations and diagram assume a non salient motor where $L_d = L_q$ which is the case of the targeted PMSM, the figures and formulas can be extended to show the relation in voltage drop for a salient PMSM where $L_d \neq L_q$.

Looking at equation 5.5, the back EMF voltage is a function of the rotor angle. It is this characteristic that the Back EMF observer uses and is able to establish and identify the rotor angle.

5.3 The Back EMF observer algorithm

Looking at equation 5.6 it is possible to note that most of the parameters and system properties shown are known values as shown in blue:

$$\begin{bmatrix} V_\alpha \\ V_\beta \end{bmatrix} = R_s * \begin{bmatrix} i_\alpha \\ i_\beta \end{bmatrix} + L_s * \frac{d}{dt} \begin{bmatrix} i_\alpha \\ i_\beta \end{bmatrix} + k_E \omega * \begin{bmatrix} -\sin(\theta) \\ \cos(\vartheta) \end{bmatrix} \quad \text{Eq 5.7}$$

V_α, V_β are known because these are calculated within the controller as an output of the current control loop, they are effectively the voltage demand in alpha and beta axis.

R_s is a known value as the motor stator resistance is a key parameter for a motor that is used within a drive. In the case of the HEMAS motor this is equal to 0.23 Ω as per Table 5.1.

i_α, i_β is a known quantity, they are captured using Analog to Digital Converters (ADCs).

L_s is the stator inductance and is a known value within a motor drive. In the case of the HEMAS motor this is equal to 1.193 mH as per Table 5.1.

k_E is a motor parameters and is a known value within a motor drive. In the case of the HEMAS motor this is equal to 0.092 V/rads as per Table 5.1.

The values that are known but can be calculated by the controller using equation 5.6 are:

$$-\omega * \sin(\theta) \quad \text{Eq 5.8}$$

and

$$\omega * \cos(\theta) \quad \text{Eq 5.9}$$

To obtain the rotor angle θ, the controller keeps calculating the internal equivalent circuit of the motor as shown in equation 5.7 and uses the two calculated quantities 5.8, 5.9 feeding an atan block therefore calculating the rotor angle. The block diagram of the observer can be seen in Figure 5.4.

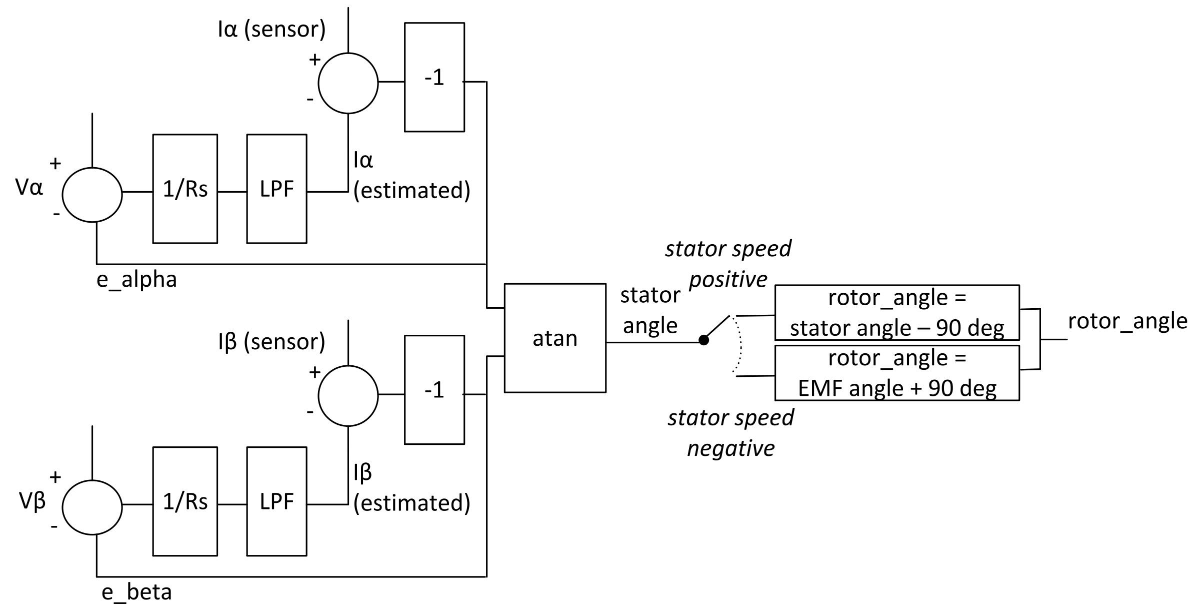


Figure 5.4 Back EMF observer block diagram

Functionality of the observer: The observer attempts to emulate the function of the motor based on its equivalent circuit as shown in Figure 5.1. If a given voltage is applied to the motor's terminals its stator resistance and inductance will result into currents and voltages as per equation 5.6. One of the primary inputs to the observer is the controller calculated voltage demands in alpha and beta axis V_α, V_β that will result into currents in the controller's equivalent circuit calculation model. The Low Pass Filter in the block diagram of Figure 5.4 emulates the LPF that is created when placing a resistor in series with an inductor. The $1/R_s$ step relates to a scaling between supplied voltage and current due to Ohm's law. If there was no back EMF voltage the estimated currents $I_{\alpha(estimated)}, I_{\beta(estimated)}$ would align with the currents from the sensors $I_{\alpha(sensor)}, I_{\beta(sensor)}$. In practice though these values would differ when the motor is spinning due to the back EMF. The observer therefore calculates this error between estimated and actual currents that is proportional to the back EMF. It is this error that is fed to the atan block that calculates the rotor angle.

Note1: the observer's atan is not fed with the $\sin(\theta)$, $\cos(\theta)$ but $-\omega * \sin(\theta)$, $\omega * \cos(\theta)$. So as the speed increases, the back EMF increases and atan calculation becomes more accurate. This is why there is typically a minimum speed threshold over which the calculation accuracy of the observer is usable.

Note2: Also because the atan is fed with $-\omega * \sin(\theta)$, $\omega * \cos(\theta)$ the atan results for positive speed into $\theta\_atan = \text{atan}(\sin(-\theta)\ ,\cos(-\theta))$ so to revert to $\theta = \theta_{atan} - 90deg$. When speed is negative $\theta\_atan = \text{atan}(\sin(-\theta)\ ,\cos(-\theta))$ so to revert to $\theta = \theta_{atan} + 90deg$.

Note3: The observer assumes constant motor parameters. In practice motor parameters may vary as temperature or operating condition changes. There is existing research [69] into online motor parameter estimation aiming to calculate the motor resistance R_s and inductance L_s. However, one of the key strengths of the observer algorithm presented in this chapter is that parameter variations or inaccuracies affect equally both alpha and beta axes. Additionally, as the estimated back EMF voltages in alpha and beta feed an atan calculation, what is of interest is the ratio of the two voltages and not their absolute values. Motor parameter variations and inaccuracies have thus minimal effect in the calculation accuracy.

Note4: the electrical time constant T of the motor is T= L_s/R_s = 5.187 ms.

5.4 The observer within the overall proposed hybrid method

Model based observers and the method described in this chapter function only when the motor is spinning. The observer is therefore only used above a defined speed threshold as shown in Figure 5.5.

To enable sensorless control at wide speed range, a hybrid model/saliency based observer is proposed. As model based methods only function when the motor is spinning, at standstill and low speed, a saliency based method is used to accelerate the motor as illustrated in Figure 5.5. This saliency based method based on High Frequency Injection (HFI) is detailed in chapter 6 of this thesis. When the motor has accelerated enough so that the back EMF voltage is of high enough amplitude for the model based observer to be accurate, the angle/speed estimation from the BEMF observer is used to control the motor.

A hysteresis method is also proposed so as to prevent continuous transitions between model and saliency based methods when approaching the transition speed. If a hysteresis region was not used, considering that some noise may exist in the estimated speed, the controller would switch between the saliency and model based method for a number of times unnecessarily until the motor speed picks up well above the transition point.

One more issue to point out that is visible in Figure 5.5 is that if speed demand changes first from 0 to +max_speed and then to -max_speed then the sensorless control mode will transition as per below

- saliency based method accelerating motor to transition speed

- model based method reaching from transition speed to +max_speed

- saliency based method while decelerating motor and then accelerating to negative speed transition point

- model based method reaching from transition negative speed to -max_speed

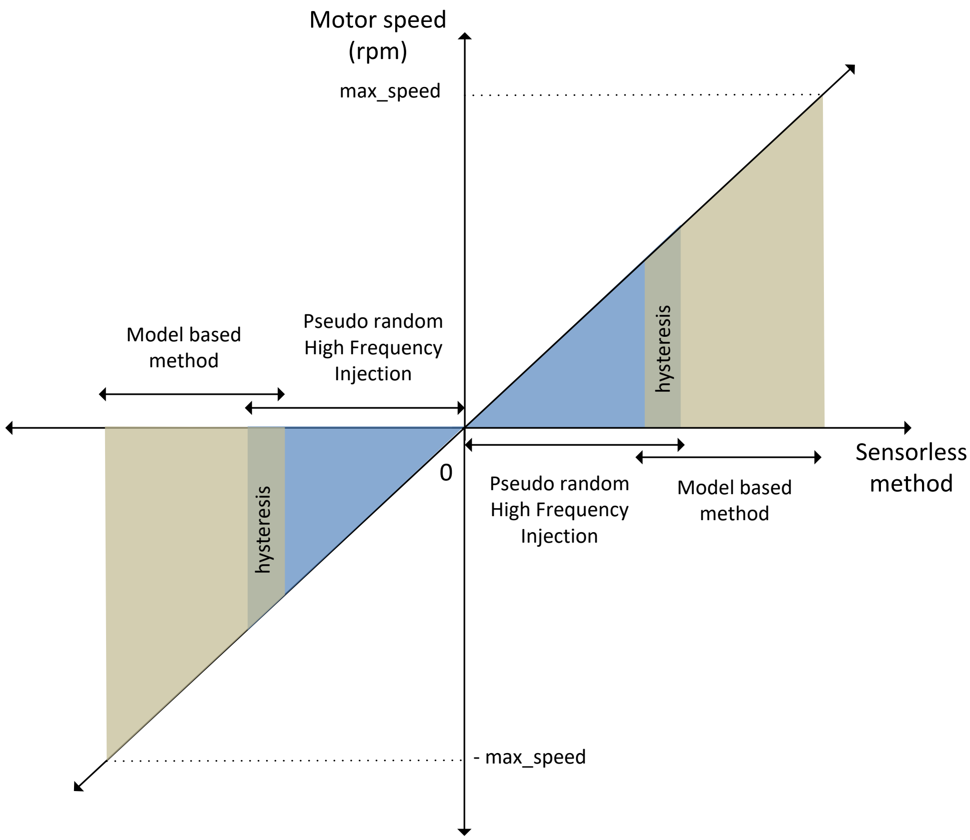


Figure 5.5 Back EMF observer block diagram

The location of the BEMF observer within the motor controller is shown in Figure 5.6.

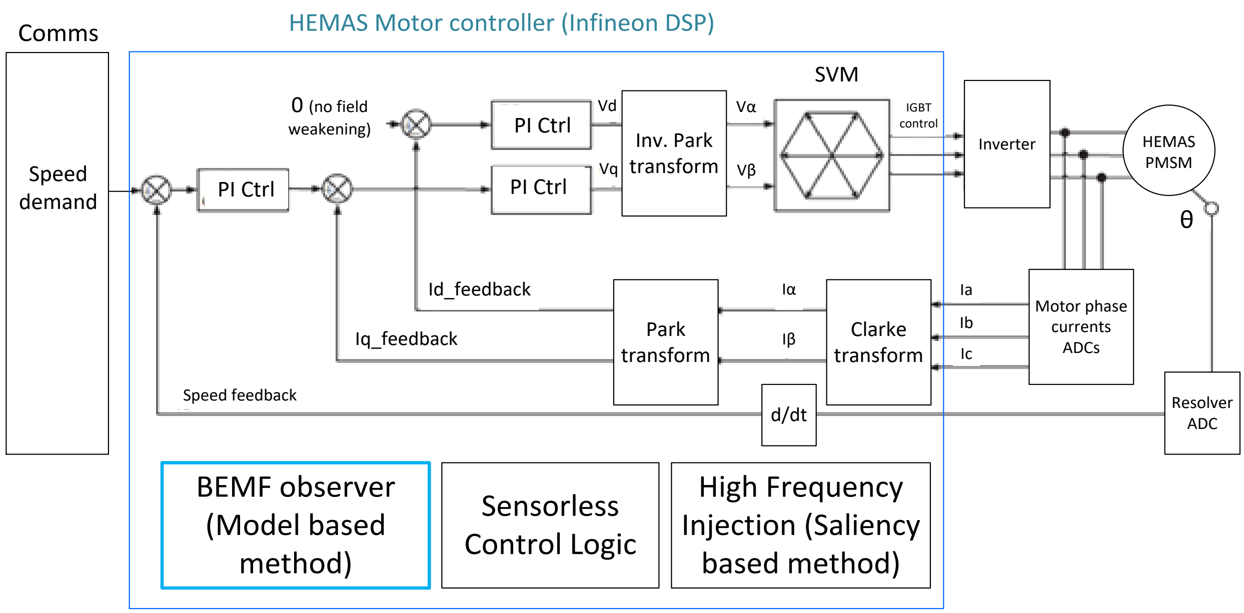


Figure 5.6 Location of BEMF observer within the overall controller

5.5 VHDL simulation of the BEMF observer

The Back EMF observer algorithm was simulated in a VHDL simulation environment at low level of implementation. The simulation included capturing data from ADC sensors at clock level, a resolver model, processing data within the controller including speed and current control loops, forward/inverse Clarke/Park transforms, Space Vector Modulation (SVM), PWM switching, an inverter and motor model. Note that the motor model parameters align with the HEMAS parameters. Simulation involved accelerating the motor from standstill using High Frequency Injection and then switching to model based observer.

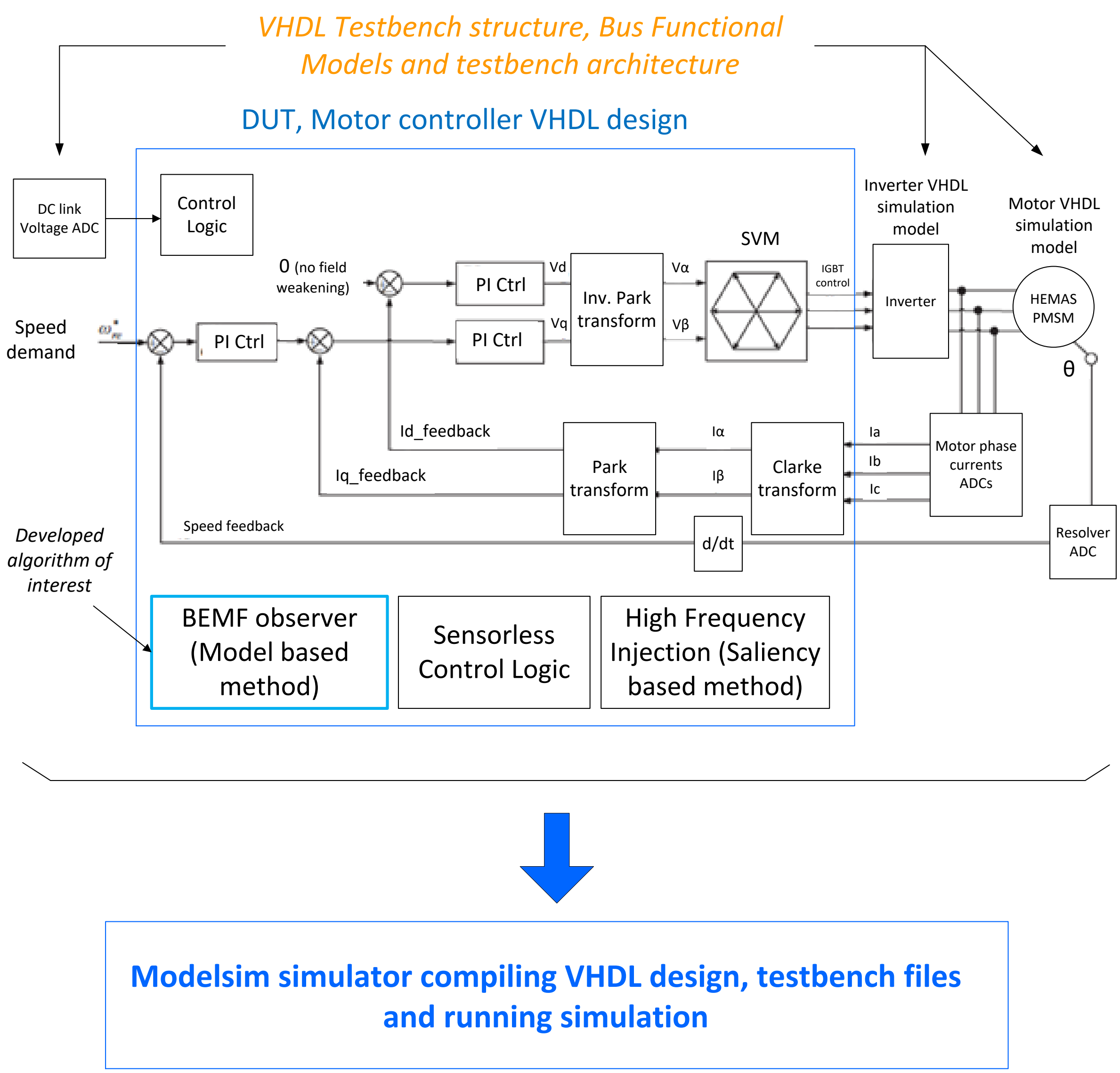


Figure 5.7 VHDL simulation structure

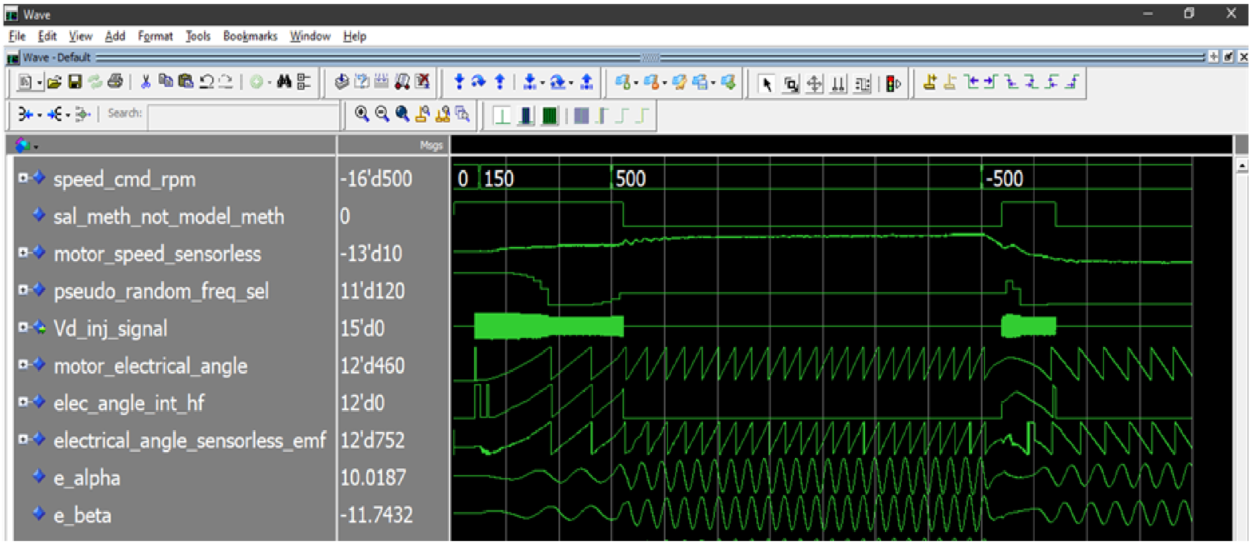


Figure 5.8 VHDL simulation zoomed out view

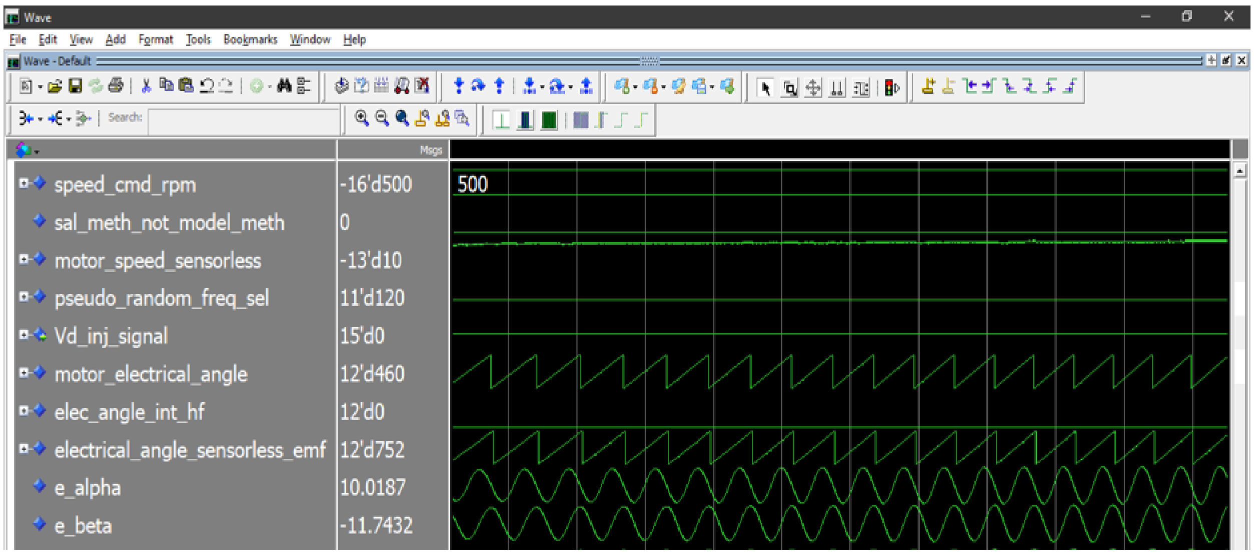


Figure 5.9 VHDL simulation zoomed in Back EMF observer

The purpose of this simulation is to evaluate the performance of the algorithm at low level.

Figure 5.58 illustrates how the motor BEMF observer tracks the electrical angle after the motor has been accelerated (note the High to Low transition of flag `sal_meth_not_model_meth`). The Motor speed command is indicated by signal `speed_smd_rpm`. Note that while the motor is stationary or the speed is not high enough for the BEMF to function, a saliency-based method is used that will be detailed in Chapter 6 of this thesis. The motor is commanded from positive to negative speed illustrating how the BEMF observer and saliency based methods are interchanged depending on the motor speed.

Figure 5.59 is a zoomed in version of 5.58 illustrating the `motor_electrical_angle`, the BEMF angle (`electrical_angle_sensorless_emf`), and the back EMF voltages (`e_alpha`, `e_beta`).

5.6 Implementation of the algorithm on the HEMAS platform

The BEMF observer algorithm analysed and simulated in previous sections was implemented in the C language targeting the Infineon DSP controller of the HEMAS platform that is used to control the swash plate of a helicopter. The BEMF observer algorithm was found to be have a very compact implementation in C, and mapped onto the targeted DSP along with debugging registers. The implementation C code of the algorithm is provided in Appendix B.2.1 of this thesis.

5.7 Hardware testing on the HEMAS platform

The section below details hardware testing results of the BEMF model based observer on the HEMAS platform.

The observer algorithm was evaluated by logging internal state of DSP registers:

- motor phase currents (see Fig 5.10 indicated in blue)

- DSP calculated resolver angle from resolver feedback signals (see Fig 5.10 indicated in blue)

- DSP calculated resolver speed from resolver feedback signals (see Fig 5.10 indicated in blue)

- BEMF observer angle

- BEMF observer speed

to a *.csv file.

The logged csv files were then imported in Excel and displayed in the following sections.

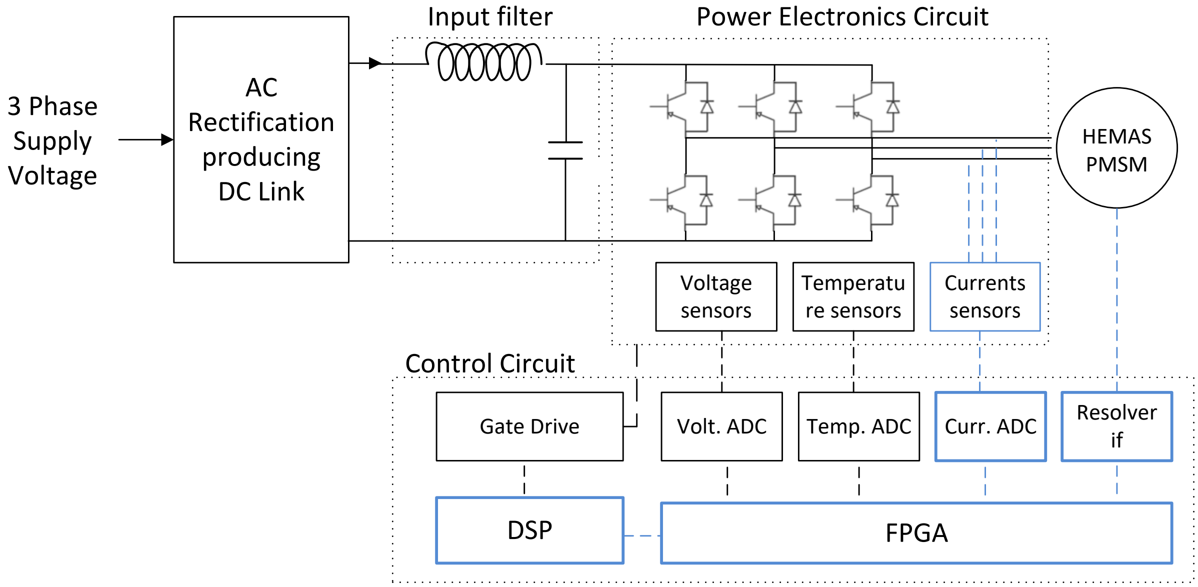


Figure 5.10 HEMAS system diagram and interfaces of interest

<u>Motor spinning with no load at 30Hz (188.5rad/sec) electrical, 360rpm mechanical</u>

Note: Sampling of variables was performed at 100 microseconds period, total of 1000 samples per variable.

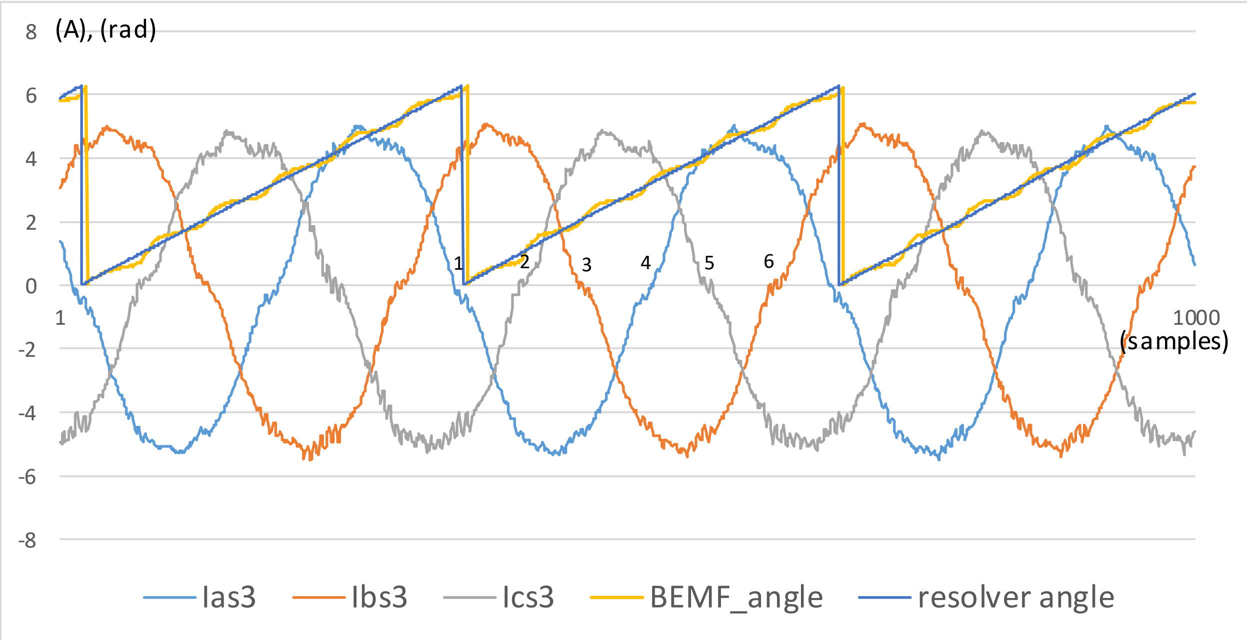


Figure 5.11 Motor spinning at 30Hz electrical (no load), observer/resolver electrical angle, phase currents

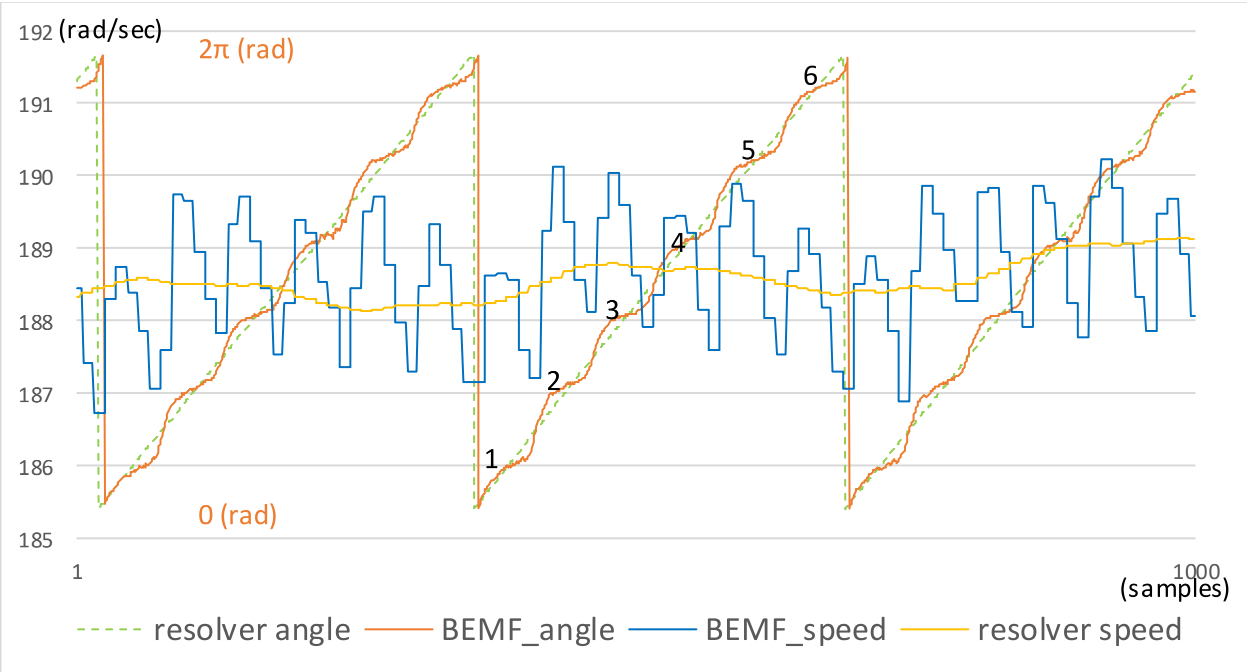


Figure 5.12 Motor spinning at 30Hz electrical (no load), observer/resolver speed (rad/sec electrical), observer/resolver electrical angle (y axis range for angle signals is 0 to 2π)

Analysis of test results 30Hz electrical with no load:

1. It can be noticed that the BEMF observer is able to track the motor's electrical angle and speed at 30Hz electrical as per Figures 5.11 - 5.12. Some harmonics are observed that are analysed in points 2 and 3 below.

2. The observer's angle suffers from harmonics at 6 times the fundamental electrical frequency as numbered in Figures 5.11, 5.12. This harmonic content in the angle estimation of the observer is due to the harmonics that also exist in the motor phase currents at the same frequency. As the BEMF observer calculates the rotor angle by means of processing the motor phase currents, the harmonics that exist in the motor phase currents propagate to the estimated angle and speed. These harmonics are due to a phenomenon known as Dead Time Distortion (DTD). The effect of DTD into the currents can be first confirmed visually by means of comparing the motor phase currents in Figure 5.11 with equivalent Figure 6a of [70] and Figure 11 of [71] also suffering from DTD. This visual observation can also be confirmed by performing an FFT to the motor phase currents of the HEMAS drive as shown in Figure 5.13. The FFT showed in fact that there is a 5th and 7th harmonic content that results into harmonic at six times the electrical frequency. This $5^{th}/7^{th}$ harmonic content aligns with the DTD harmonic contents presented in [70] and [71]. Note that the underlying cause of the non-sinusoidal shape of the currents that propagates to the observer calculation is a delay introduced between switching of the upper and lower IGBTs in an inverter leg aiming to prevent DC link shorts. This delay has a visible impact on the motor phase currents when they transition through zero i.e. 6 times within an electrical period. This distortion makes the currents of less sinusoidal shape as shown in Figure 5.11 points numbered 1 to 6. The distortion in the motor phase currents due to DTD also results into increased torque ripple and acoustic noise as torque is a function of currents (see Equation 5.1). As the sensorless observer is using the motor phase currents to calculate the rotor angle, the estimated electrical angle also experiences this harmonic at 6 times the electrical frequency.

3. Figure 5.12 illustrates the comparison between motor's speed as calculated by resolver interface versus the sensorless observer. The harmonic in the observer's electrical angle analysed in point 2 above is amplified in the observer speed as the speed is the derivative of the angle and is therefore sensitive to sudden changes and oscillations. Note that the observer's speed ripple takes place 6 times within each electrical period. Similar results apply for remaining spinning frequencies.

4. There is a number of ways to mitigate the issue of DTD and specifically the harmonic effect to the sensorless observer. The effect of DTD in the motor phase currents can be reduced by what is known as Dead Time Compensation (DTC). DTC involves applying additional voltage per phase to compensate for the effect of Dead Time. While DTC reduces the effects of DTD it does not eliminate them completely. To reduce the DTD further, a band stop filter can be introduced within the BEMF observer aiming to suppress these harmonics. The bandstop filter would be centered at 6 times the electrical frequency it would therefore be dynamic as the motor spins. While these two types of compensations are not the primary topic of this research will be included in the future work section of this thesis.

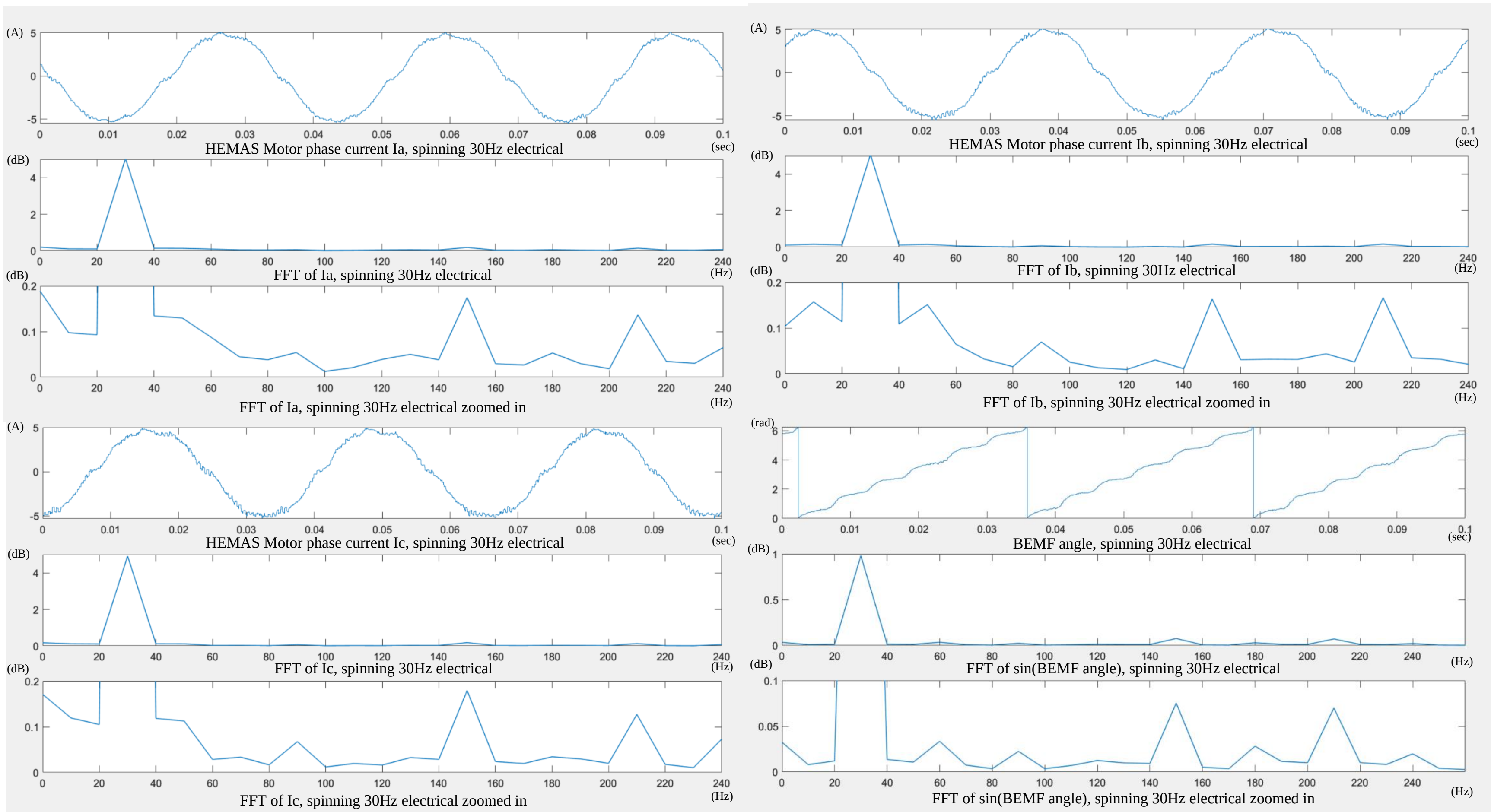


Figure 5.13 FFT analysis of motor phase currents and BEMF angle

Note that the tests following do not cover the full speed range of the motor (5200 rpm). This is because the test rig is not designed to support the full range. Instead, the range covered by the tests reaches up to 2520 rpm. As the analysis of below test scenarios is similar to the 30Hz scenario to prevent duplication, there is an overall analysis in the end following all the tests artefacts.

Motor spinning under load at 30Hz (188.5rad/sec) electrical, 360rpm mechanical

Note1: Sampling of variables was performed at 100 microseconds period, total of 1000 samples per variable.

Note2: A magnetic particle brake from Placid Industries (model PFB-400) was used to apply the load to the motor while spinning setting the voltage to 1V resulting to torque of ≈ 4 Nm.

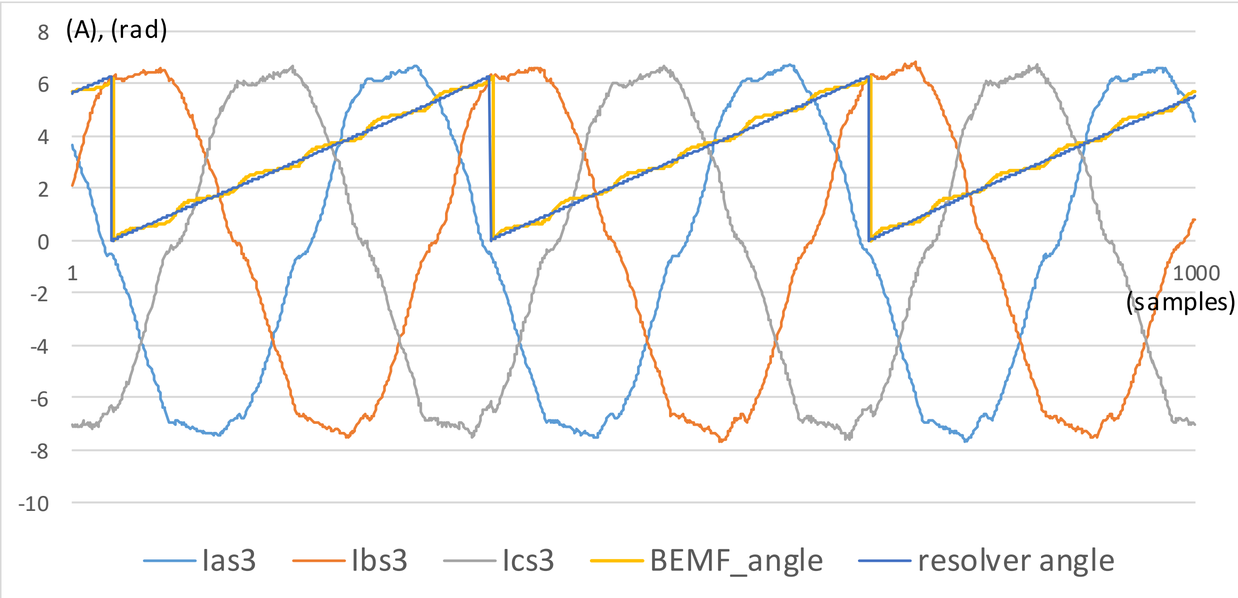


Figure 5.14 Motor spinning at 30Hz electrical (under load), observer/resolver electrical angle, phase currents

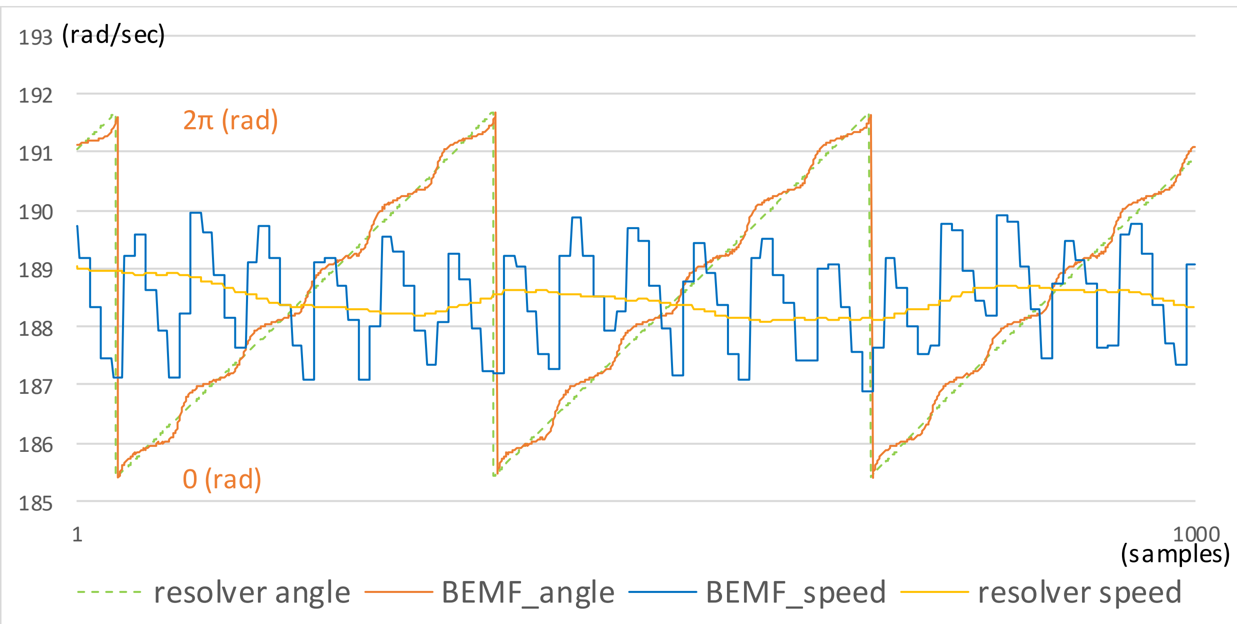


Figure 5.15 Motor spinning at 30Hz electrical (under load), observer/resolver speed (rad/sec), observer/resolver electrical angle (y axis range for angle signals is 0 to 2π)

Motor spinning with no load at 40Hz (251.3rad/sec) electrical, 480rpm mechanical

Note: Sampling of variables was performed at 100 microseconds period, total of 1000 samples per variable.

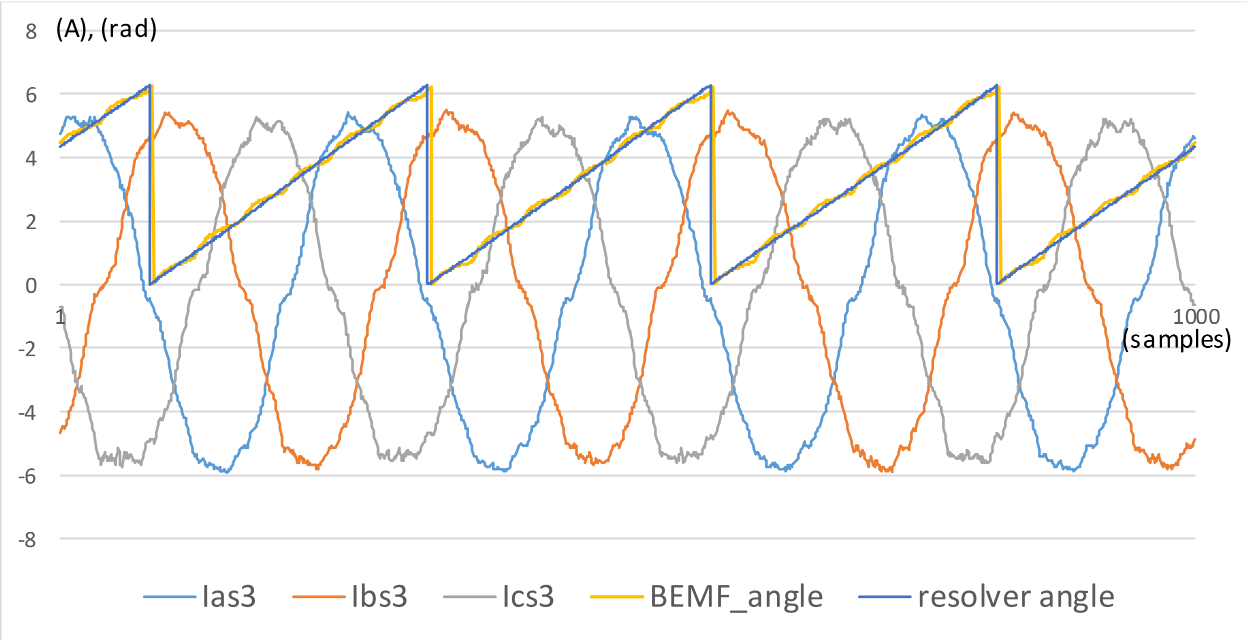


Figure 5.16 Motor spinning at 40Hz electrical (no load), observer/resolver electrical angle, phase currents

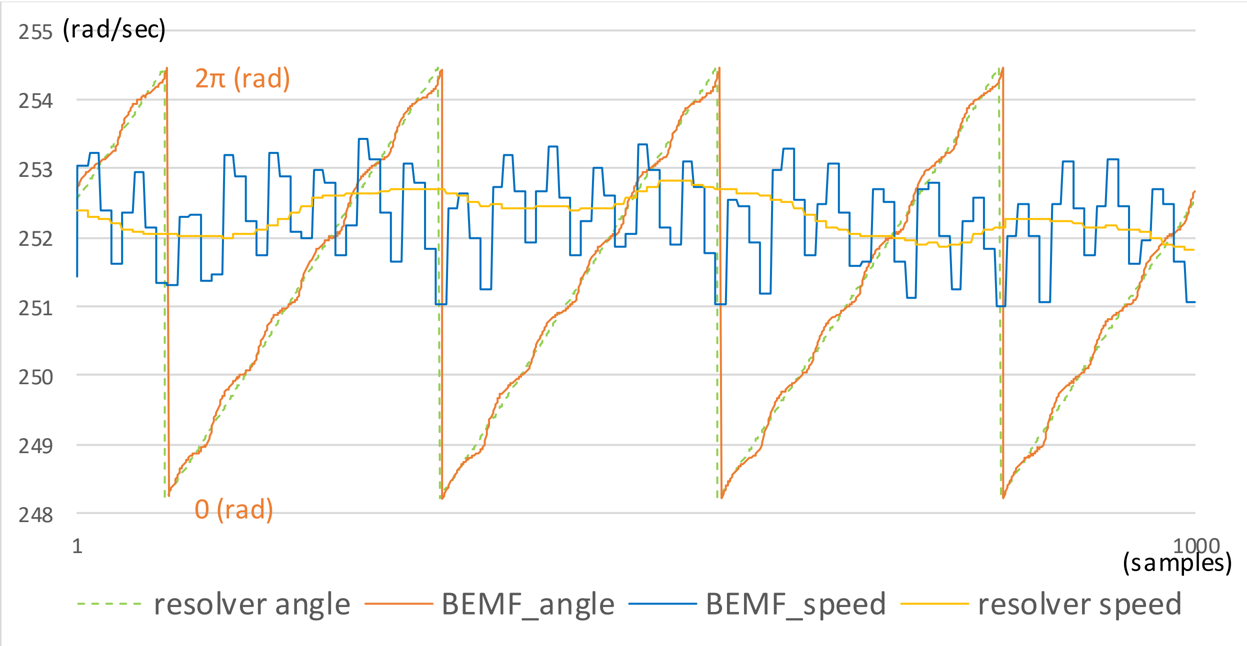


Figure 5.17 Motor spinning at 40Hz electrical (no load), observer/resolver speed (rad/sec) and electrical angle (y axis range for angle signals is 0 to 2π)

Motor spinning under load at 40Hz (251.3rad/sec) electrical, 480rpm mechanical

Note1: Sampling of variables was performed at 100 microseconds period, total of 1000 samples per variable.

Note2: A magnetic particle brake from Placid Industries (model PFB-400) was used to apply the load to the motor while spinning setting the voltage to 1V resulting to torque of ≈ 4 Nm.

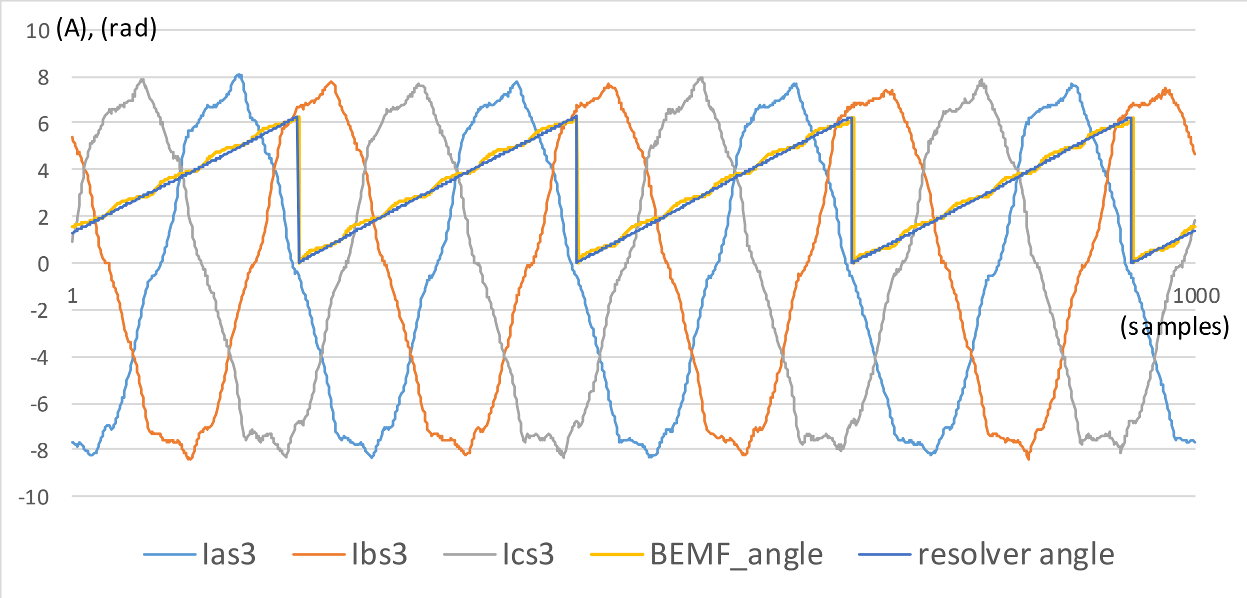


Figure 5.18 Motor spinning at 40Hz electrical (under load), observer/resolver electrical angle, phase currents

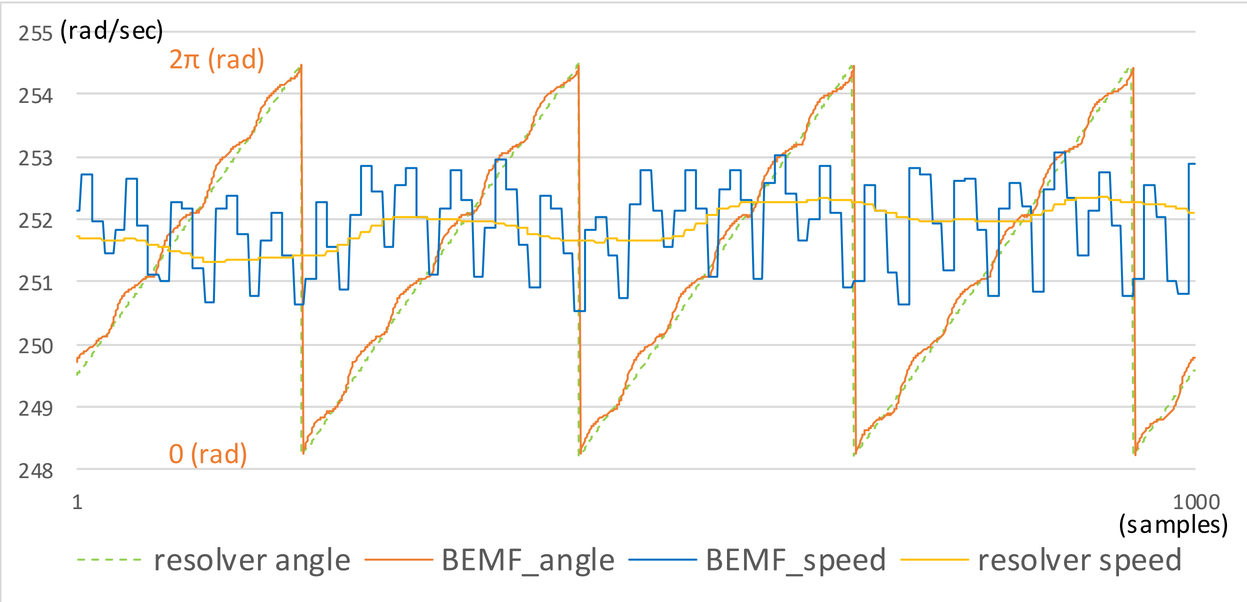


Figure 5.19 Motor spinning at 40Hz electrical (under load), observer/resolver speed (rad/sec) and electrical angle (y axis range for angle signals is 0 to 2π)

Motor spinning with no load at 60Hz (377rad/sec) electrical, 720rpm mechanical

Note1: Sampling of variables was performed at 100 microseconds period, total of 1000 samples per variable.

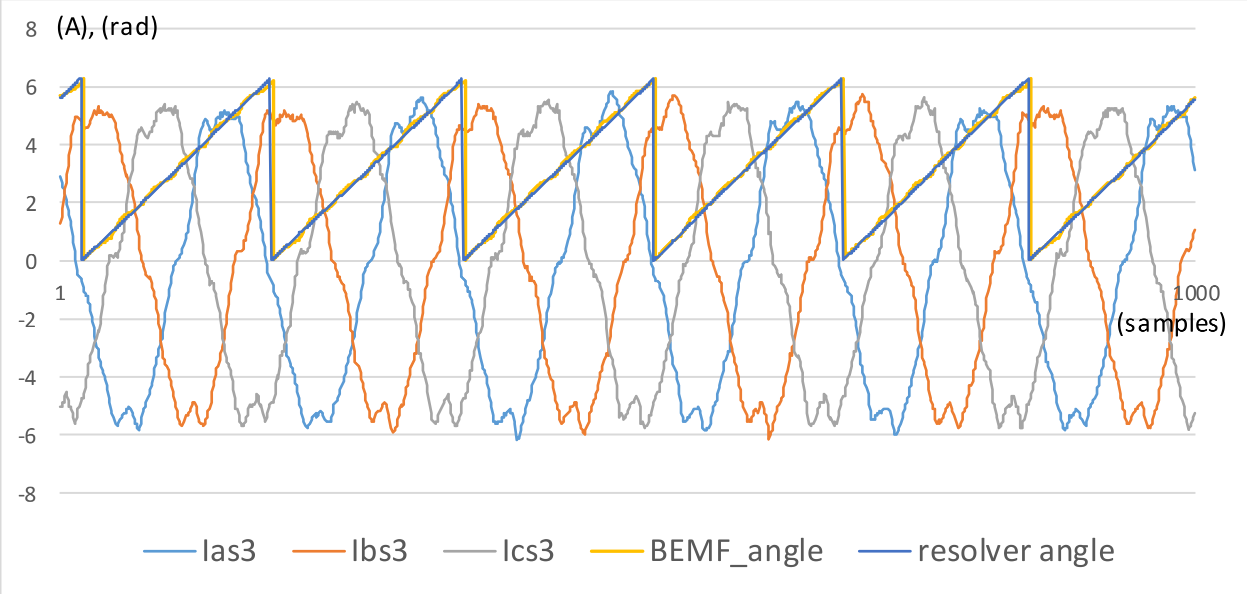


Figure 5.20 Motor spinning at 60Hz electrical (no load), observer/resolver electrical angle, phase currents

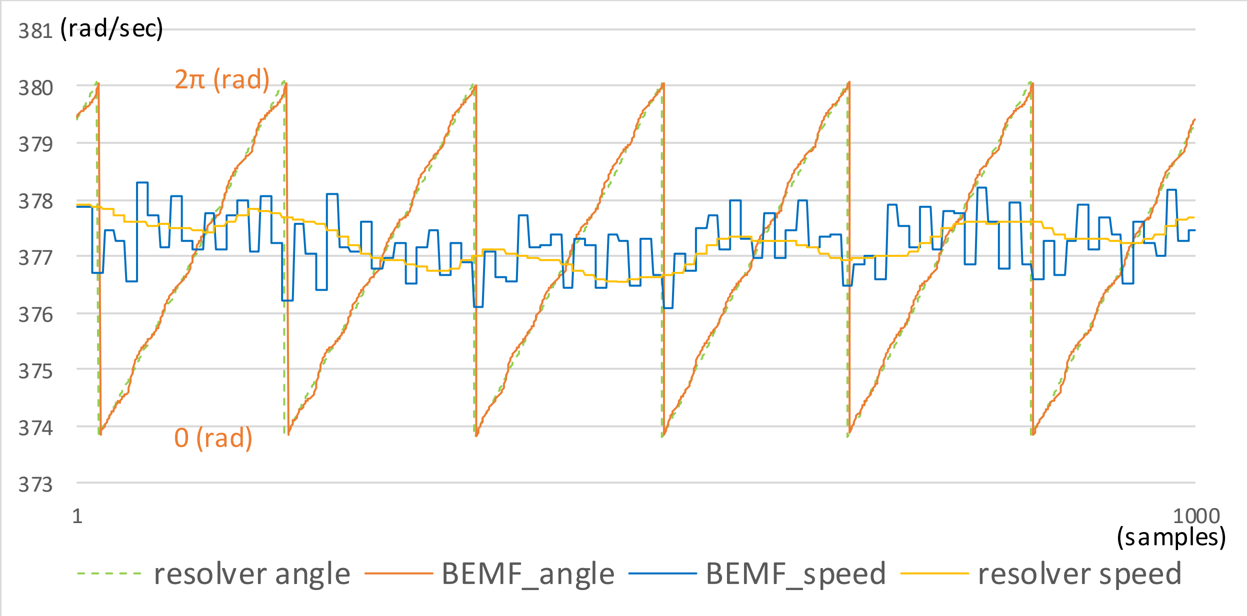


Figure 5.21 Motor spinning at 60Hz electrical (no load), observer/resolver speed (rad/sec) and electrical angle (y axis range for angle signals is 0 to 2π)

<u>Motor spinning under load at 60Hz (377rad/sec) electrical, 720rpm mechanical</u>

Note1: Sampling of variables was performed at 100 microseconds period, total of 1000 samples per variable.

Note2: A magnetic particle brake from Placid Industries (model PFB-400) was used to apply the load to the motor while spinning setting the voltage to 1V resulting to torque of ≈ 4 Nm.

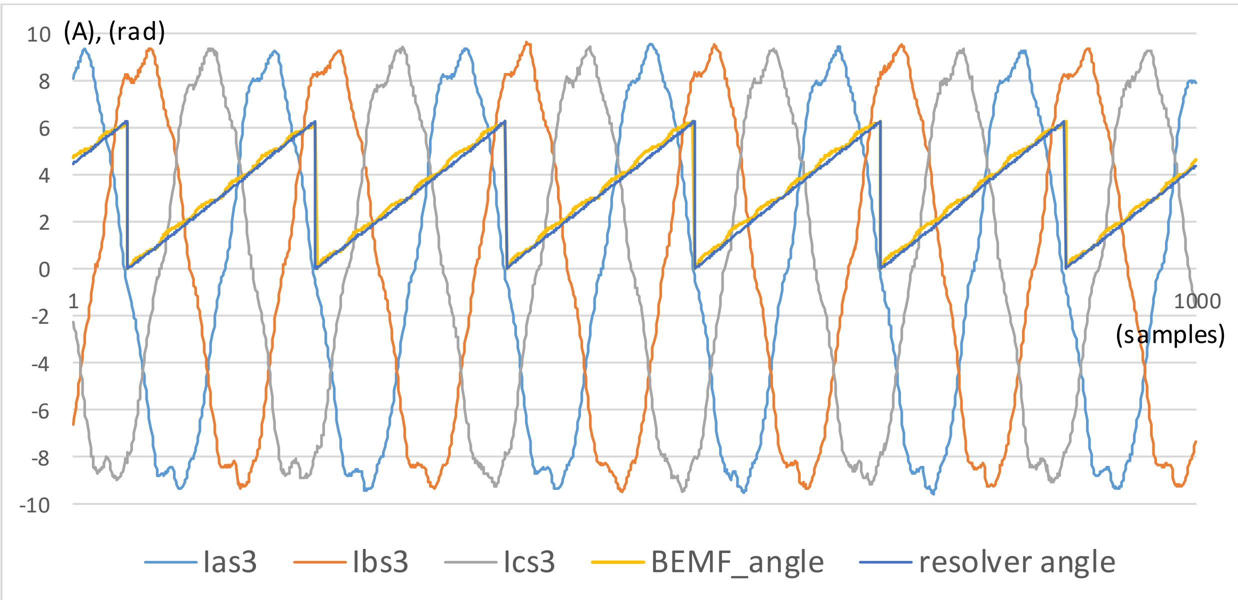


Figure 5.22 Motor spinning at 60Hz electrical (under load), observer/resolver electrical angle, phase currents

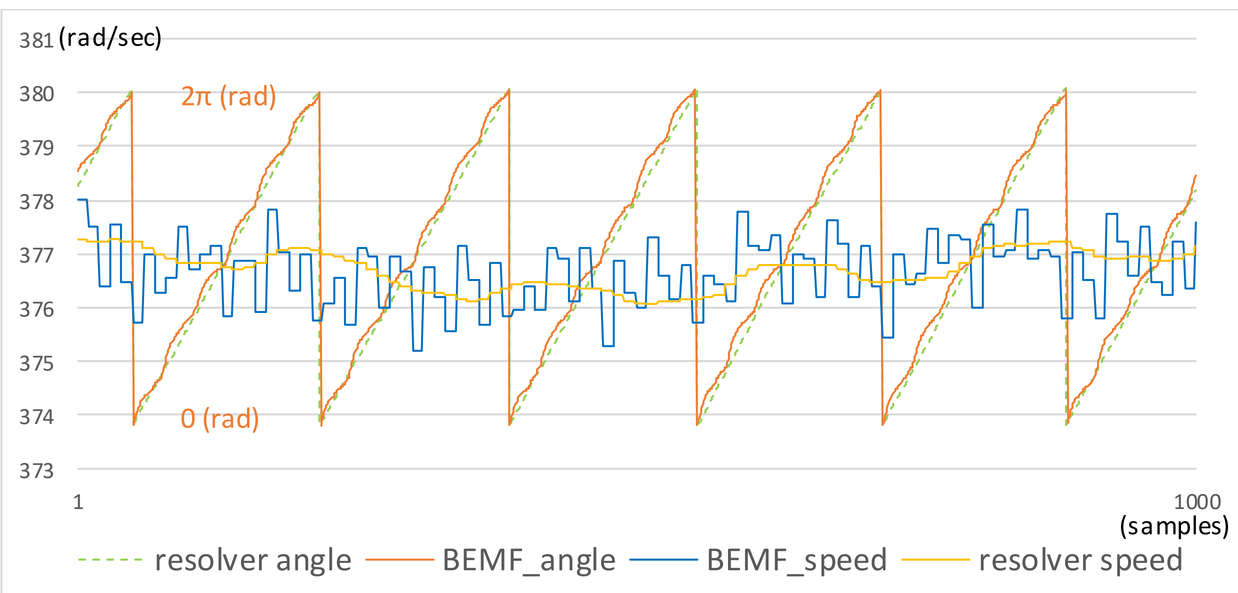


Figure 5.23 Motor spinning at 60Hz electrical (under load), observer/resolver speed (rad/sec) and electrical angle (y axis range for angle signals is 0 to 2π)

Motor spinning with no load at 80Hz (502rad/sec) electrical, 960rpm mechanical

Note: Sampling of variables was performed at 100 microseconds period, total of 1000 samples per variable.

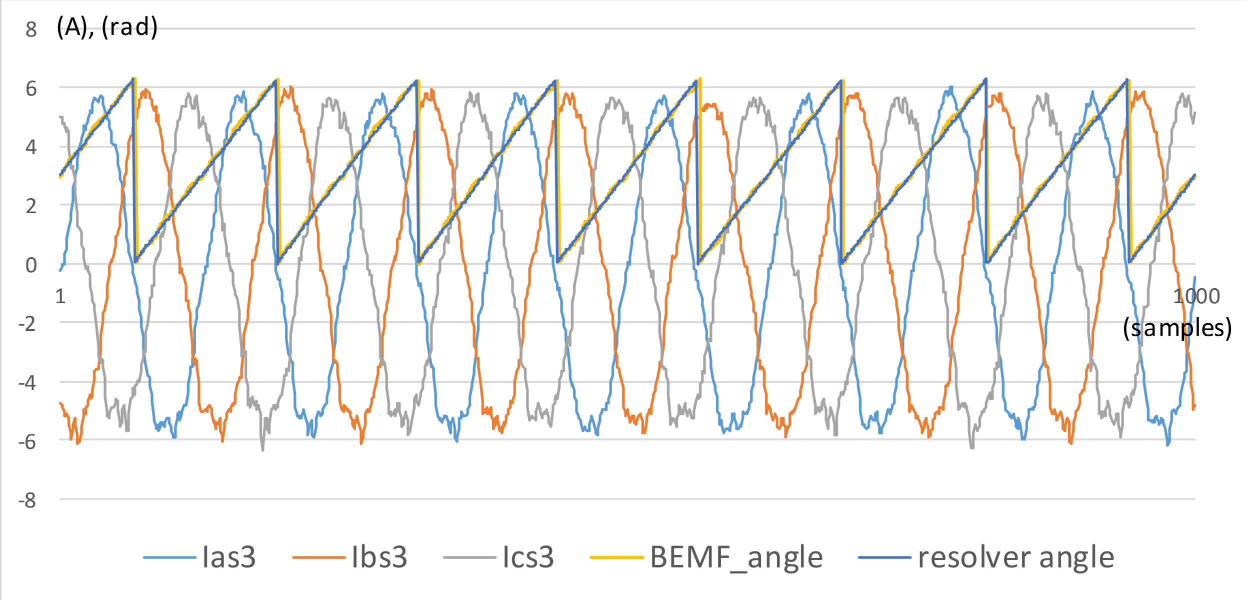


Figure 5.24 Motor spinning at 80Hz electrical (no load), observer/resolver electrical angle, phase currents

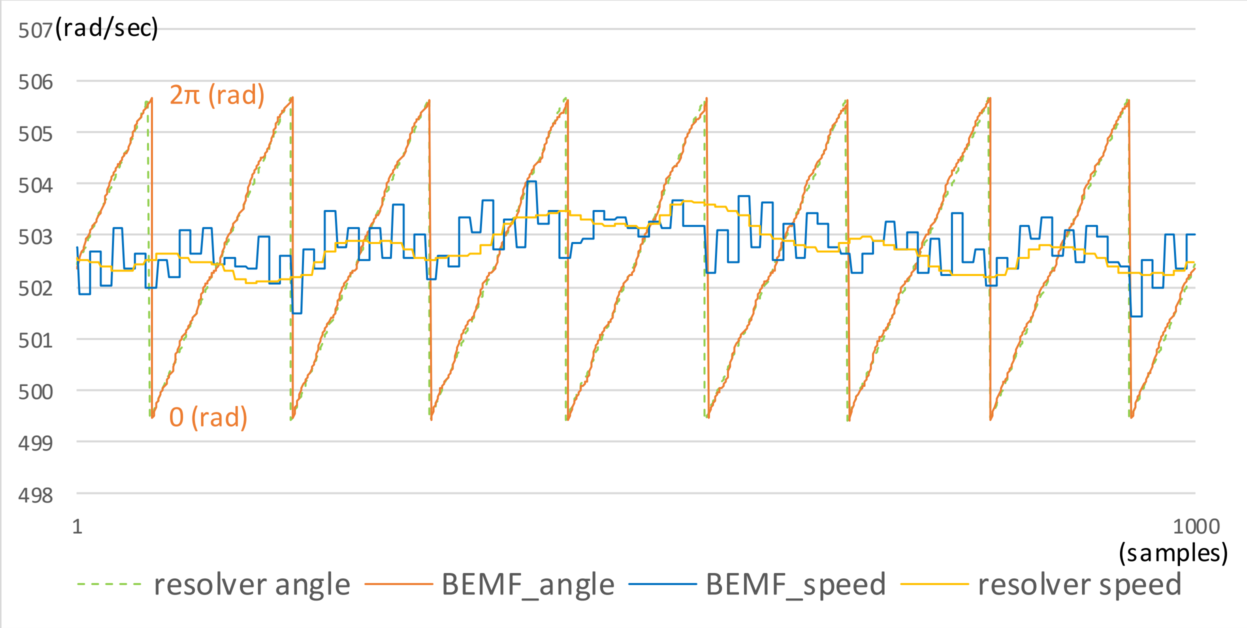


Figure 5.25 Motor spinning at 80Hz electrical (no load), observer/resolver speed (rad/sec) and electrical angle (y axis range for angle signals is 0 to 2π)

<u>Motor spinning under load at 80Hz (502rad/sec) electrical, 960rpm mechanical</u>

Note1: Sampling of variables was performed at 100 microseconds period, total of 1000 samples per variable.

Note2: A magnetic particle brake from Placid Industries (model PFB-400) was used to apply the load to the motor while spinning setting the voltage to 1V resulting to torque of ≈ 4 Nm.

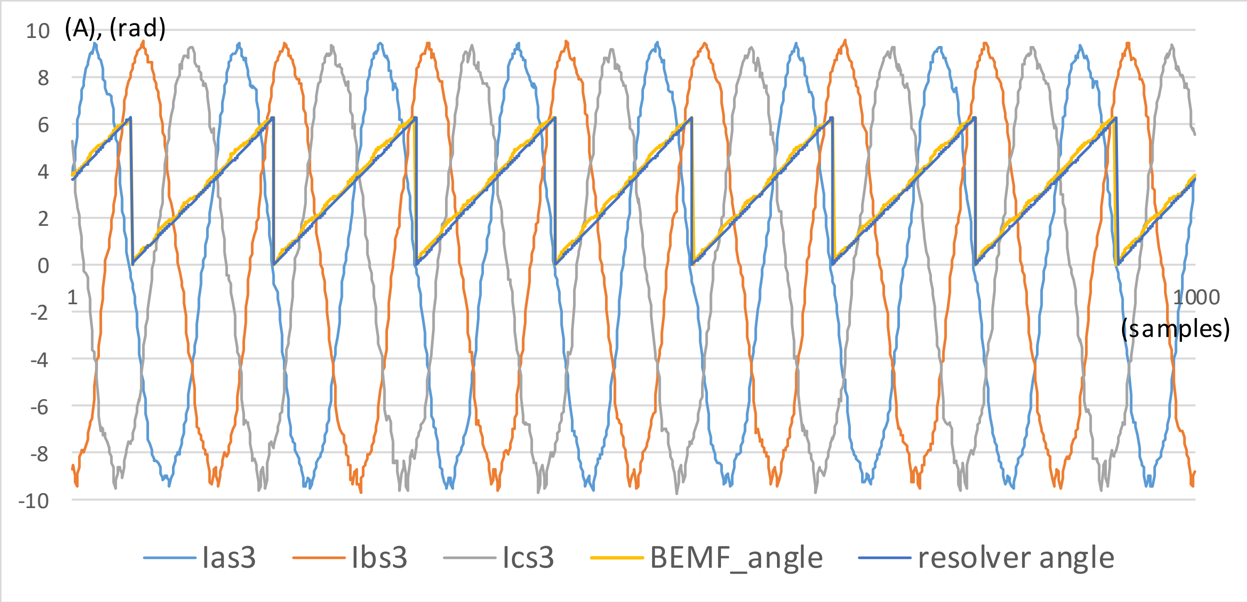


Figure 5.26 Motor spinning at 80Hz electrical (under load), observer/resolver electrical angle, phase currents

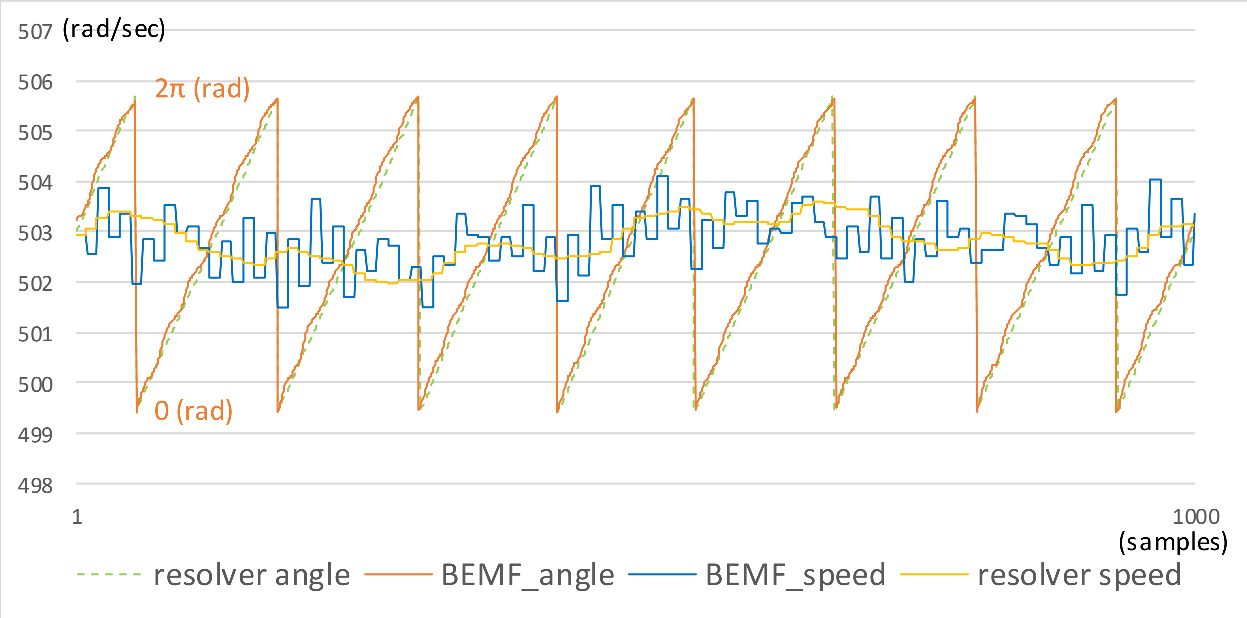


Figure 5.27 Motor spinning at 80Hz electrical (under load), observer/resolver speed (rad/sec) and electrical angle (y axis range for angle signals is 0 to 2π)

<u>Motor spinning with no load at 100Hz (628rad/sec) electrical, 1200rpm mechanical</u>

Note: Sampling of variables was performed at 100 microseconds period, total of 1000 samples per variable.

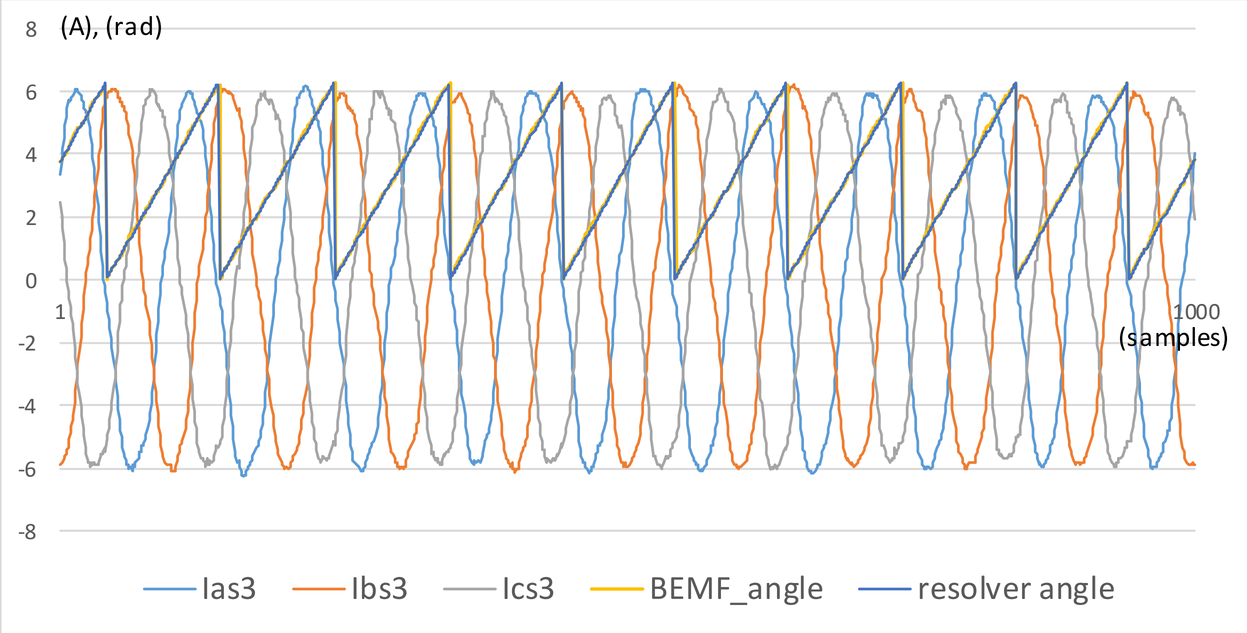


Figure 5.28 Motor spinning at 100Hz electrical (no load), observer/resolver electrical angle, phase currents

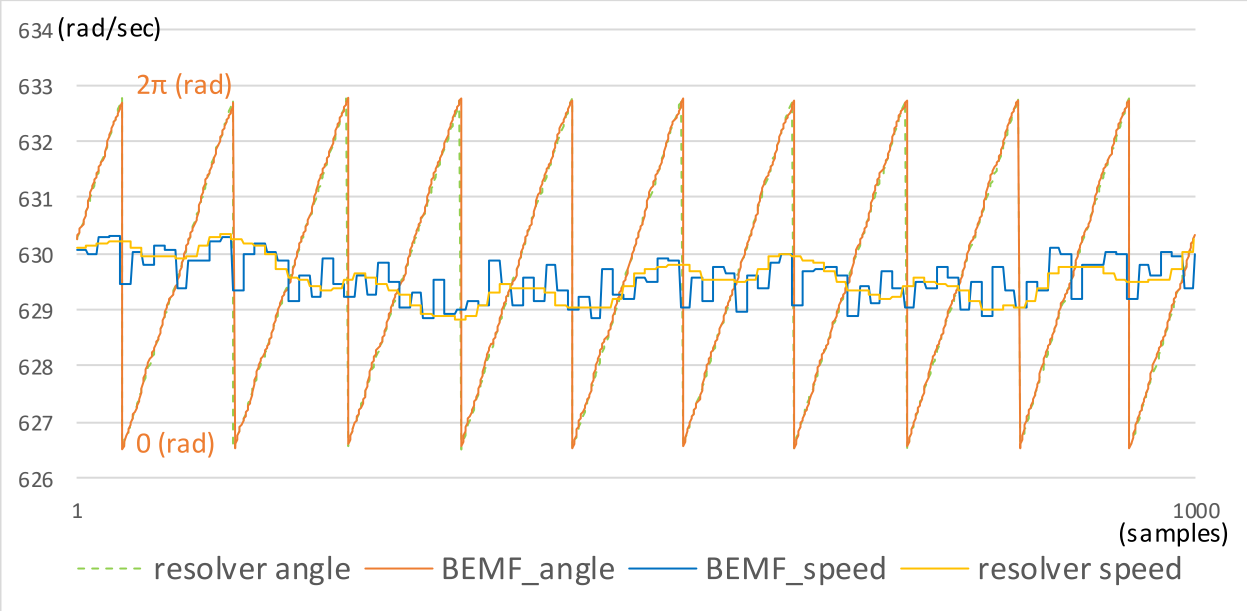


Figure 5.29 Motor spinning at 100Hz electrical (no load), observer/resolver speed (rad/sec) and electrical angle (y axis range for angle signals is 0 to 2π)

Motor spinning under load at 100Hz (628rad/sec) electrical, 1200rpm mechanical

Note1: Sampling of variables was performed at 100 microseconds period, total of 1000 samples per variable.

Note2: A magnetic particle brake from Placid Industries (model PFB-400) was used to apply the load to the motor while spinning setting the voltage to 1V resulting to torque of ≈ 4 Nm.

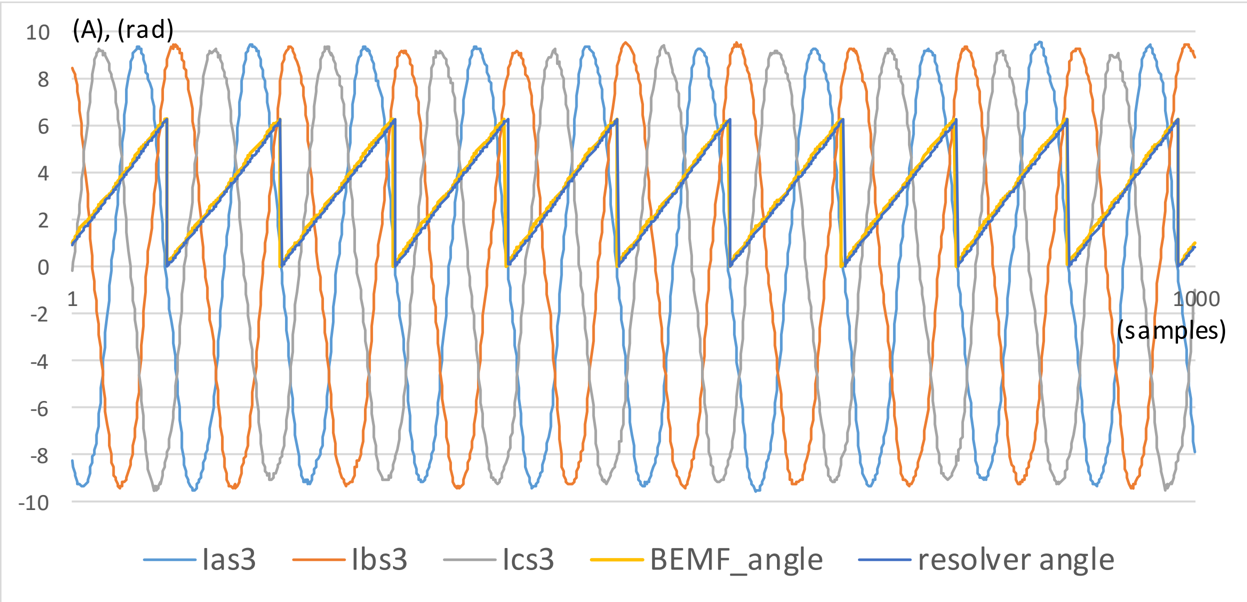


Figure 5.30 Motor spinning at 100Hz electrical (no load), observer/resolver electrical angle, phase currents

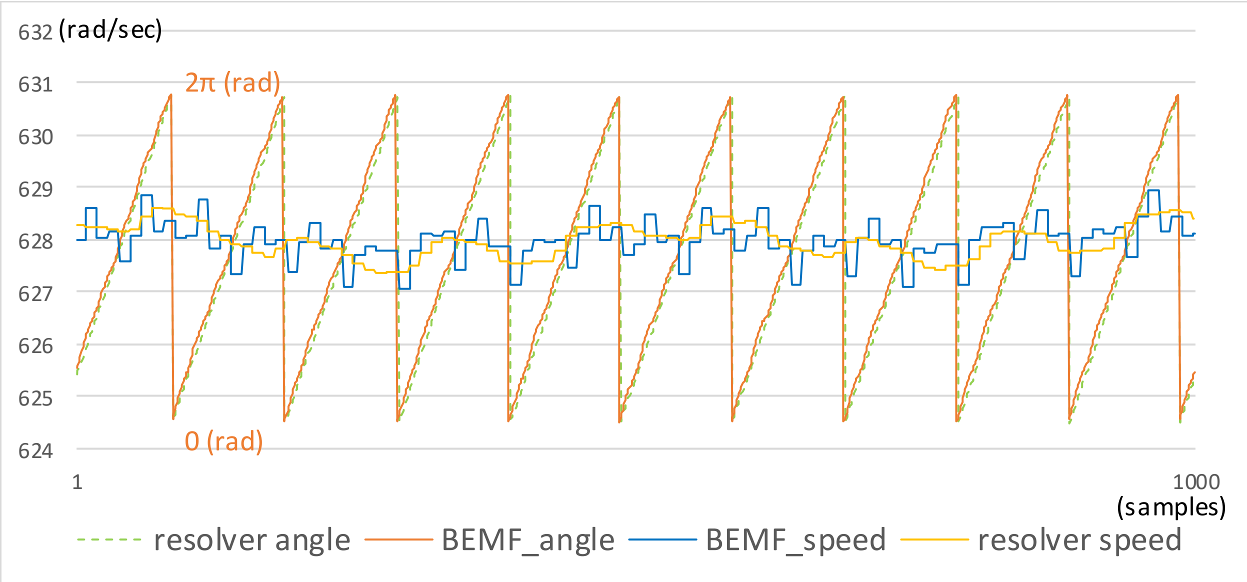


Figure 5.31 Motor spinning at 100Hz electrical (no load), observer/resolver speed (rad/sec) and electrical angle (y axis range for angle signals is 0 to 2π)

Motor spinning with no load at 120Hz (754rad/sec) electrical, 1440rpm mechanical

Note: Sampling of variables was performed at 100 microseconds period, total of 1000 samples per variable.

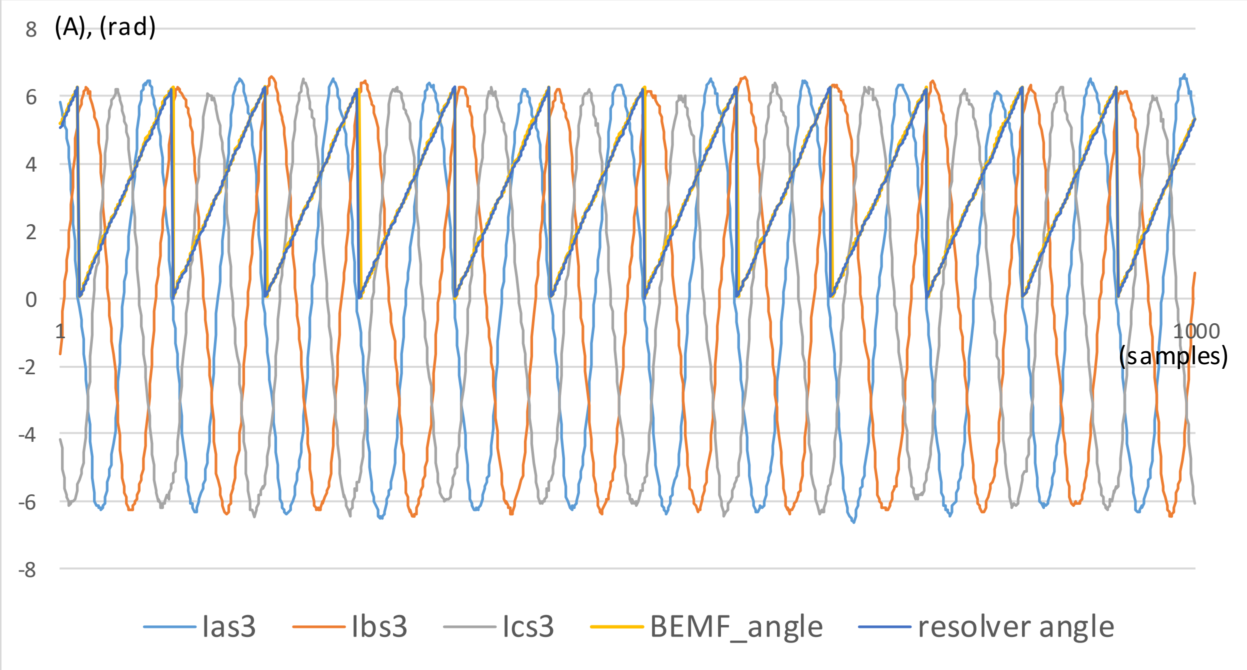


Figure 5.32 Motor spinning at 120Hz electrical (no load), observer/resolver electrical angle, phase currents

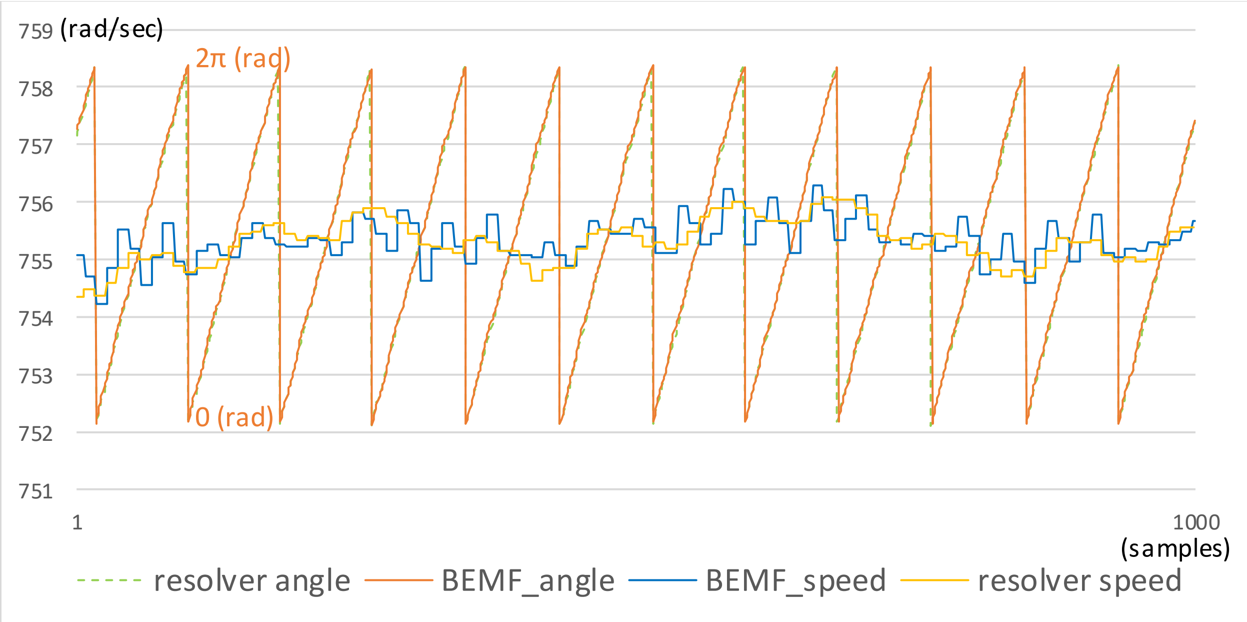


Figure 5.33 Motor spinning at 120Hz electrical (no load), observer/resolver speed (rad/sec) and electrical angle (y axis range for angle signals is 0 to 2π)

Motor spinning under load at 120Hz (754rad/sec) electrical, 1440rpm mechanical

Note1: Sampling of variables was performed at 100 microseconds period, total of 1000 samples per variable.

Note2: A magnetic particle brake from Placid Industries (model PFB-400) was used to apply the load to the motor while spinning setting the voltage to 1V resulting to torque of ≈ 4 Nm.

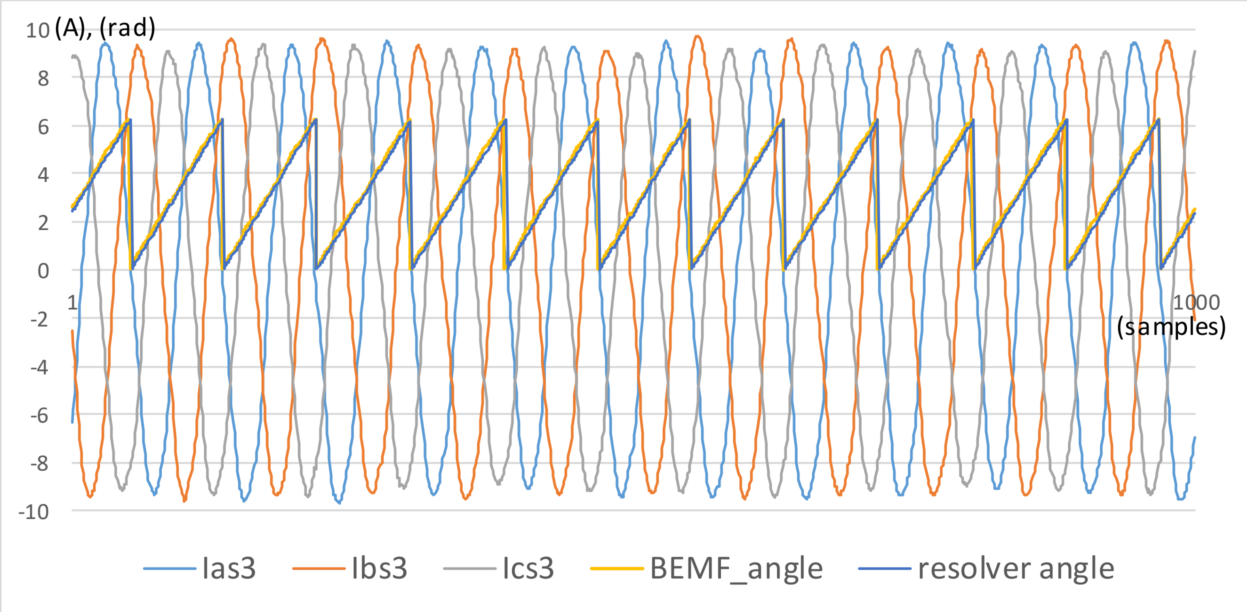


Figure 5.34 Motor spinning at 120Hz electrical (no load), observer/resolver electrical angle, phase currents

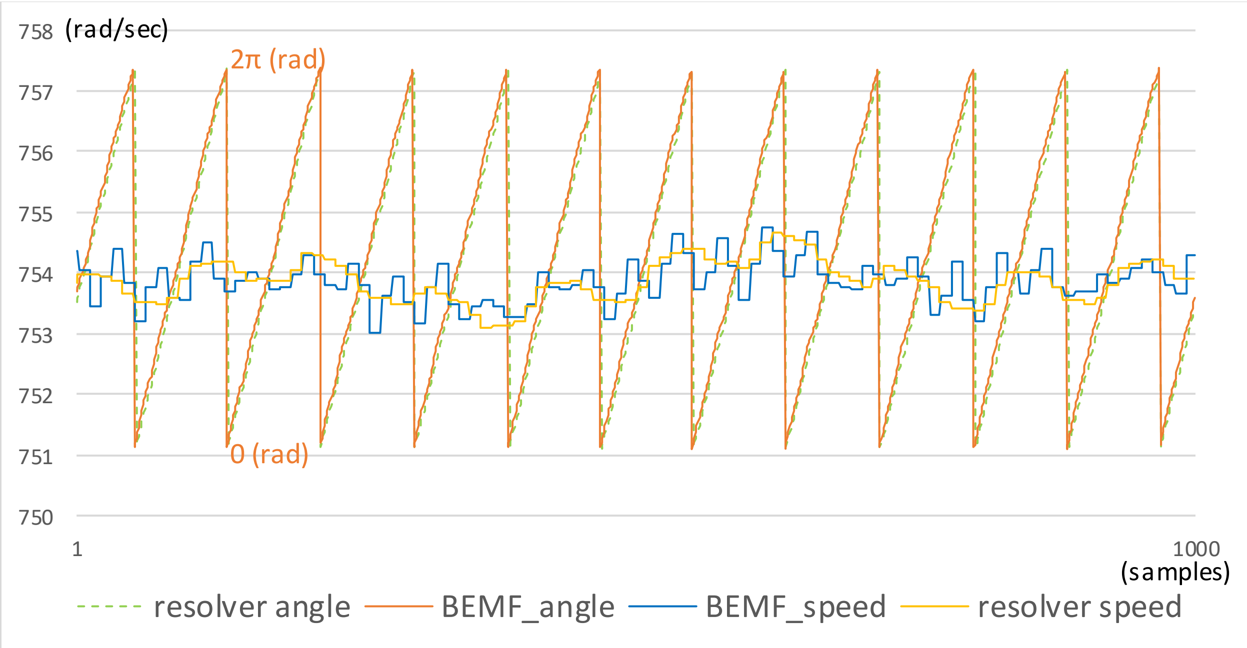


Figure 5.35 Motor spinning at 120Hz electrical (no load), observer/resolver speed (rad/sec) and electrical angle (y axis range for angle signals is 0 to 2π)

Motor spinning with no load at 140Hz (879rad/sec) electrical, 1680rpm mechanical

Note: Sampling of variables was performed at 100 microseconds period, total of 1000 samples per variable.

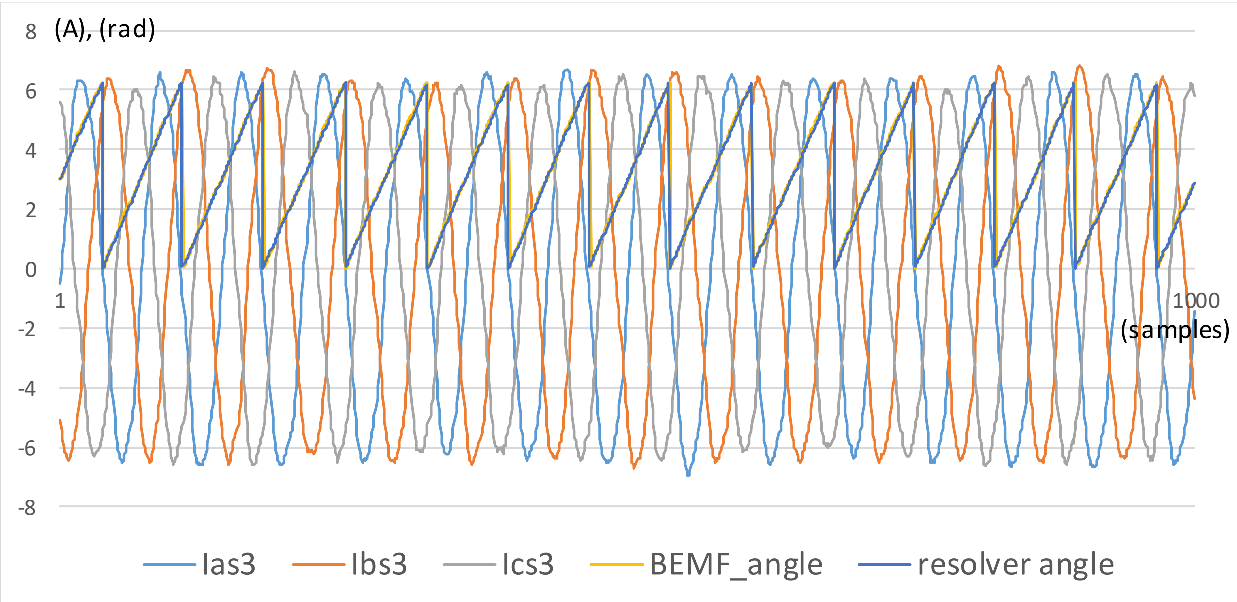


Figure 5.36 Motor spinning at 140Hz electrical (no load), observer/resolver electrical angle, phase currents

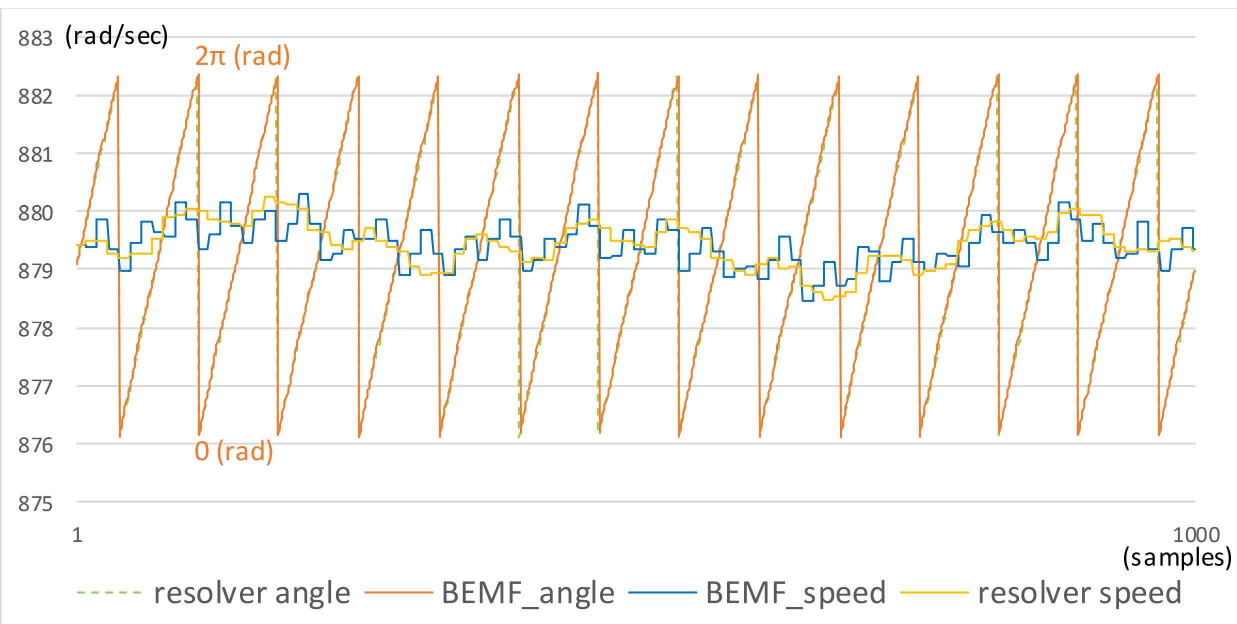


Figure 5.37 Motor spinning at 140Hz electrical (no load), observer/resolver speed (rad/sec) and electrical angle (y axis range for angle signals is 0 to 2π)

Motor spinning under load at 140Hz (879rad/sec) electrical, 1680rpm mechanical

Note1: Sampling of variables was performed at 100 microseconds period, total of 1000 samples per variable.

Note2: A magnetic particle brake from Placid Industries (model PFB-400) was used to apply the load to the motor while spinning setting the voltage to 1V resulting to torque of ≈ 4 Nm.

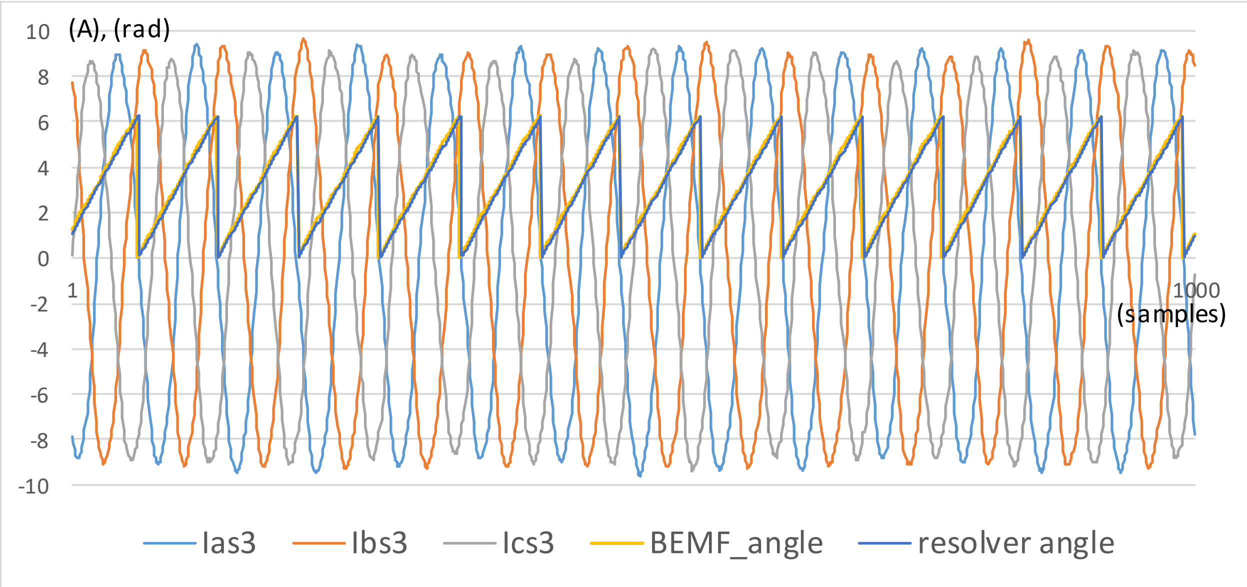


Figure 5.38 Motor spinning at 140Hz electrical (no load), observer/resolver electrical angle, phase currents

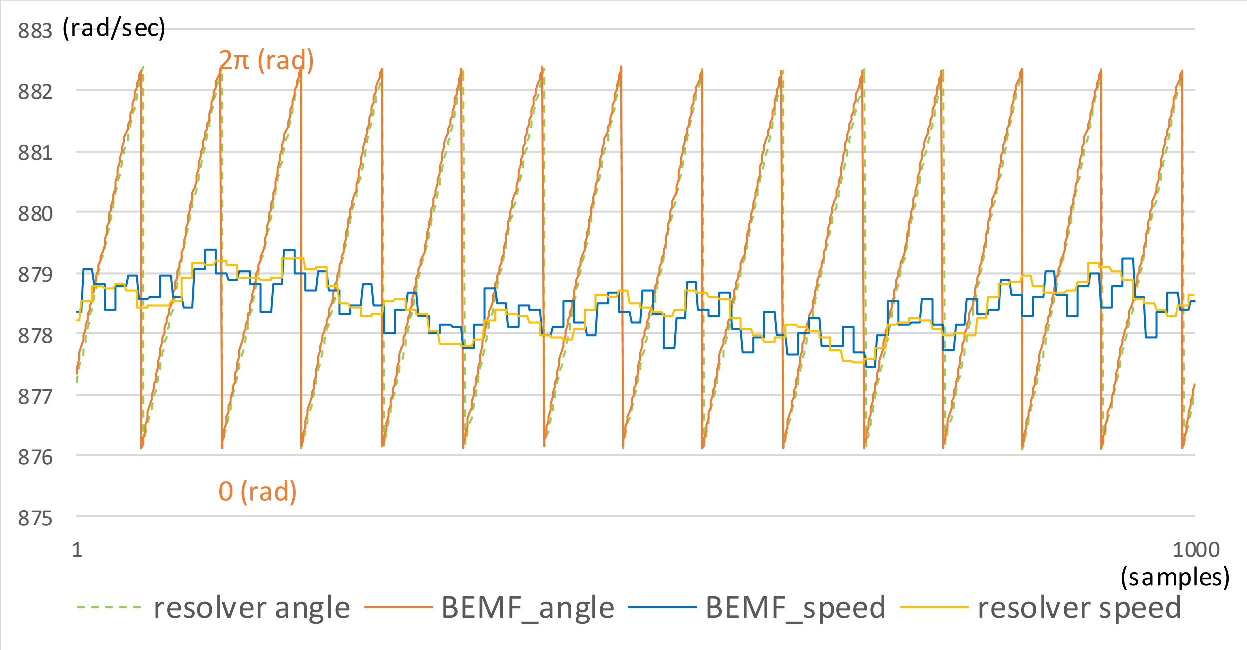


Figure 5.39 Motor spinning at 140Hz electrical (no load), observer/resolver speed (rad/sec) and electrical angle (y axis range for angle signals is 0 to 2π)

<u>Motor spinning with no load at 160Hz (1005rad/sec) electrical, 1920rpm mechanical</u>

Note: Sampling of variables was performed at 100 microseconds period, total of 1000 samples per variable.

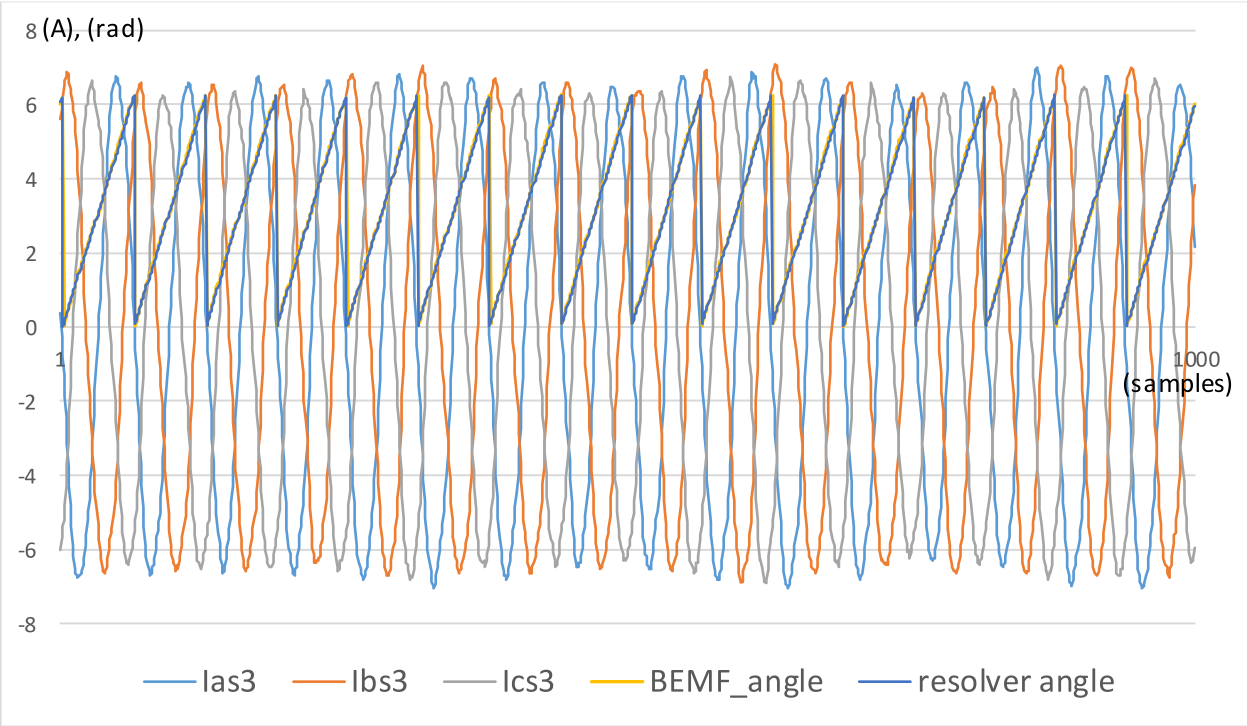


Figure 5.40 Motor spinning at 160Hz electrical (no load), observer/resolver electrical angle, phase currents

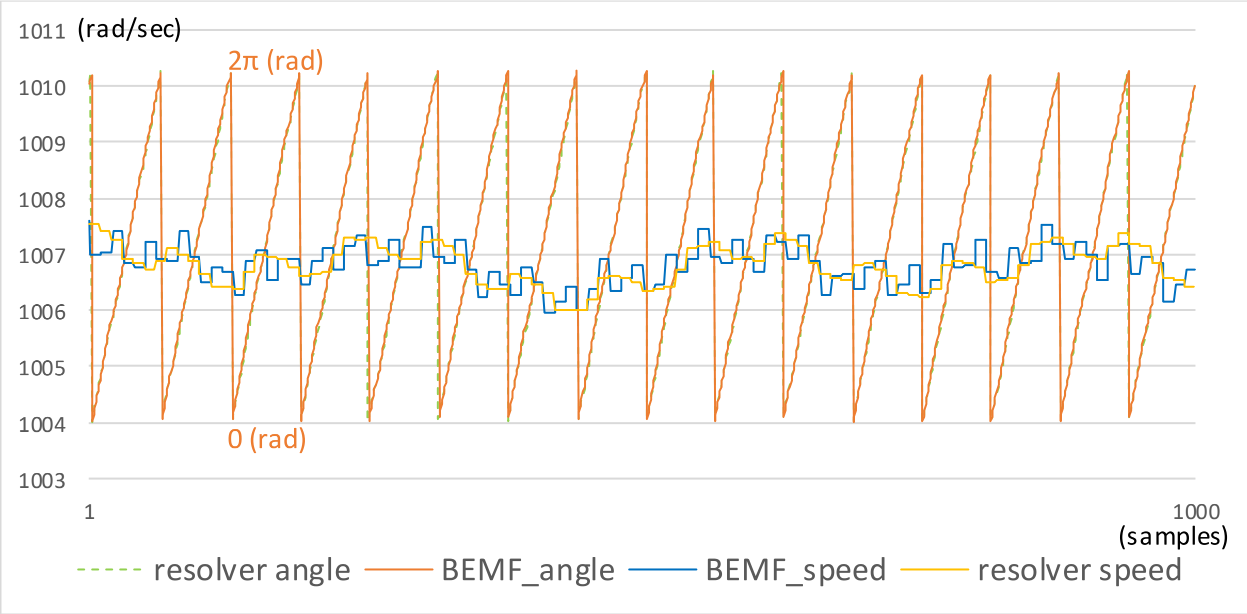


Figure 5.41 Motor spinning at 160Hz electrical (no load), observer/resolver speed (rad/sec) and electrical angle (y axis range for angle signals is 0 to 2π)

Motor spinning under load at 160Hz (1005rad/sec) electrical, 1920rpm mechanical

Note1: Sampling of variables was performed at 100 microseconds period, total of 1000 samples per variable.

Note2: A magnetic particle brake from Placid Industries (model PFB-400) was used to apply the load to the motor while spinning setting the voltage to 1V resulting to torque of ≈ 4 Nm.

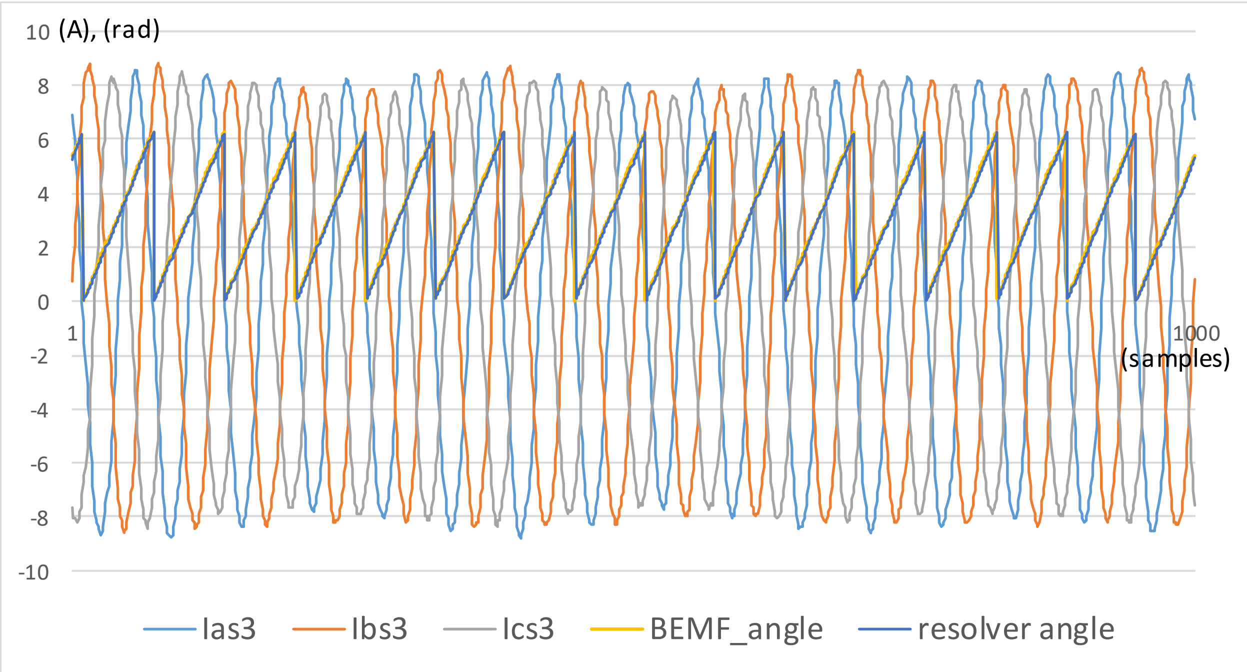


Figure 5.42 Motor spinning at 160Hz electrical (load), observer/resolver electrical angle, phase currents

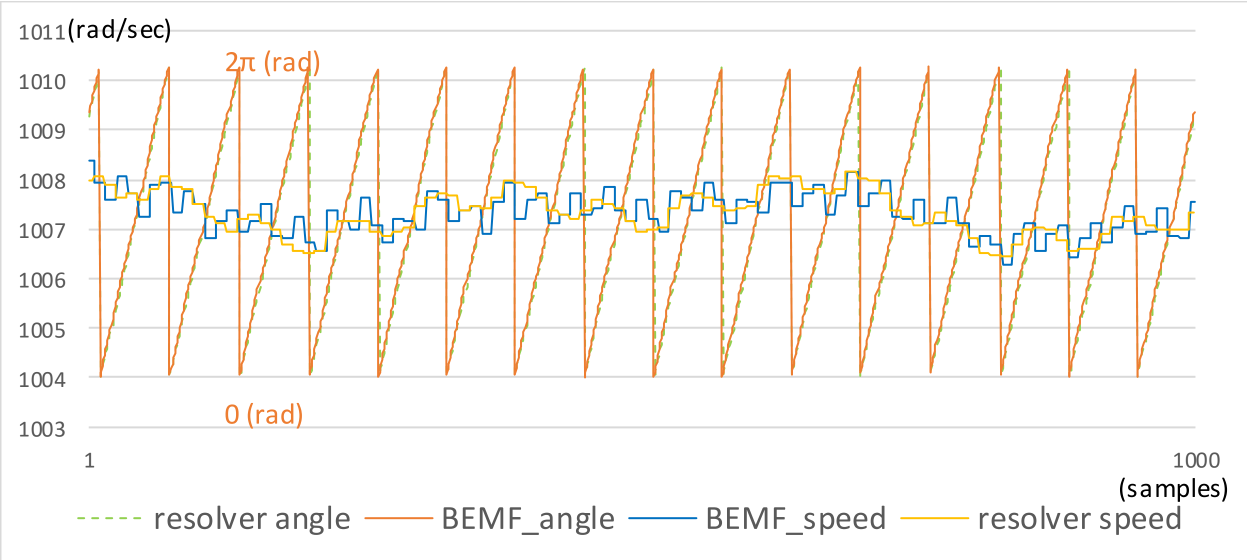


Figure 5.43 Motor spinning at 160Hz electrical (load), observer/resolver speed (rad/sec) and electrical angle (y axis range for angle signals is 0 to 2π)

Motor spinning with no load at 180Hz (1130rad/sec) electrical, 2160rpm mechanical

Note: Sampling of variables was performed at 100 microseconds period, total of 1000 samples per variable.

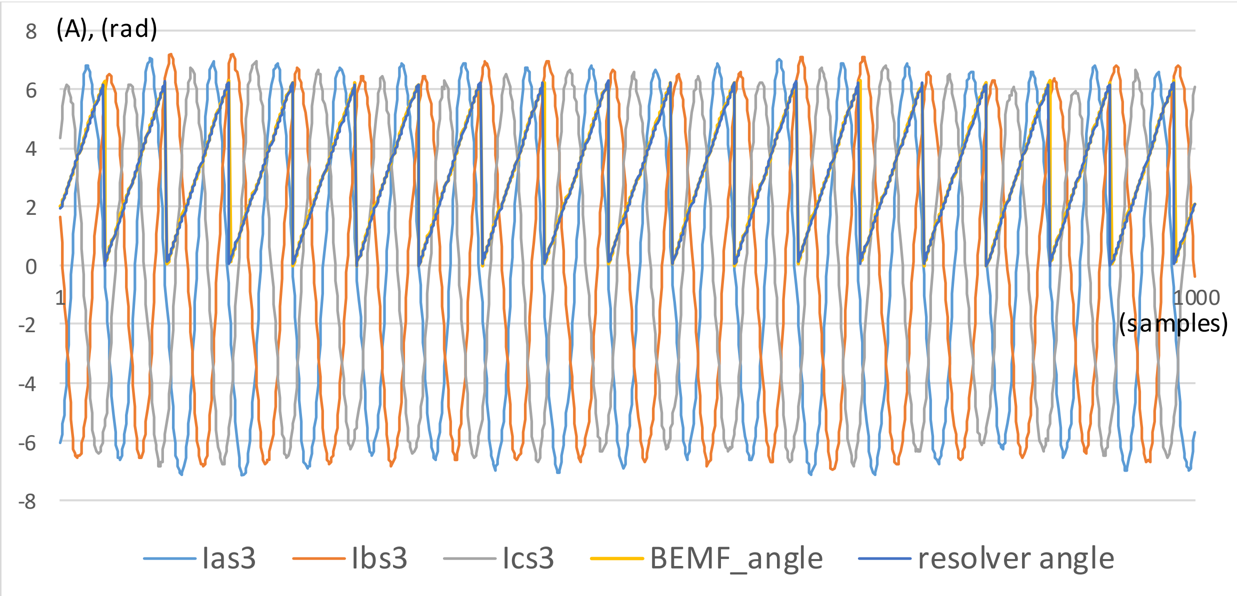


Figure 5.44 Motor spinning at 180Hz electrical (no load), observer/resolver electrical angle, phase currents

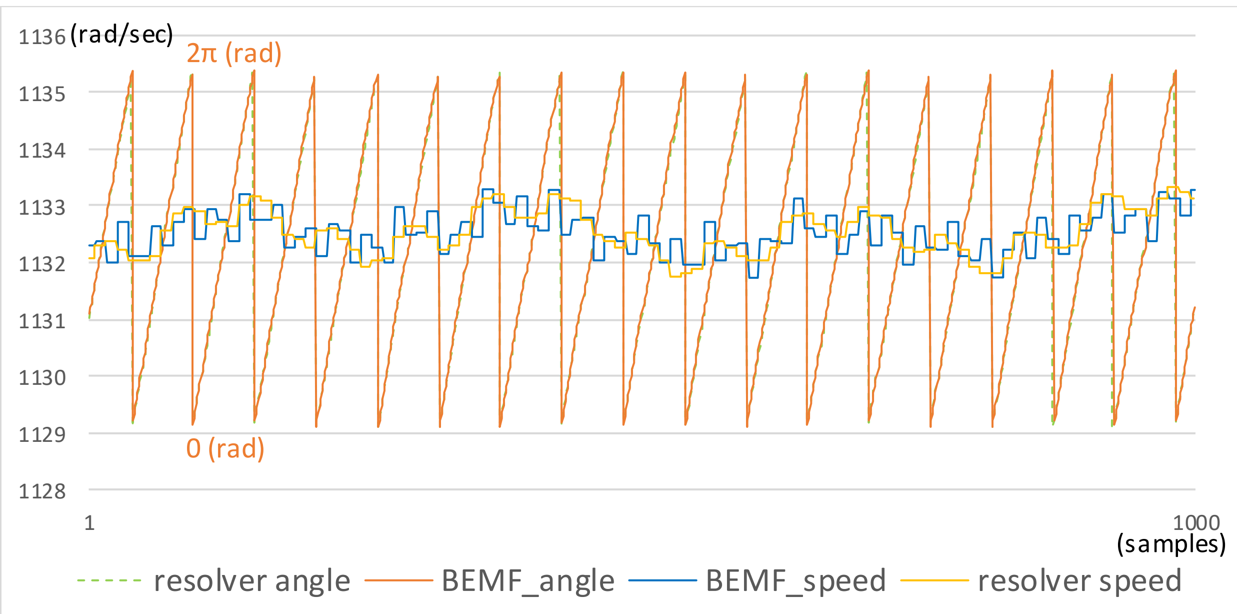


Figure 5.45 Motor spinning at 180Hz electrical (no load), observer/resolver speed (rad/sec) and electrical angle (y axis range for angle signals is 0 to 2π)

Motor spinning under load at 180Hz (1130rad/sec) electrical, 2160rpm mechanical

Note1: Sampling of variables was performed at 100 microseconds period, total of 1000 samples per variable.

Note2: A magnetic particle brake from Placid Industries (model PFB-400) was used to apply the load to the motor while spinning setting the voltage to 1V resulting to torque of ≈ 4 Nm.

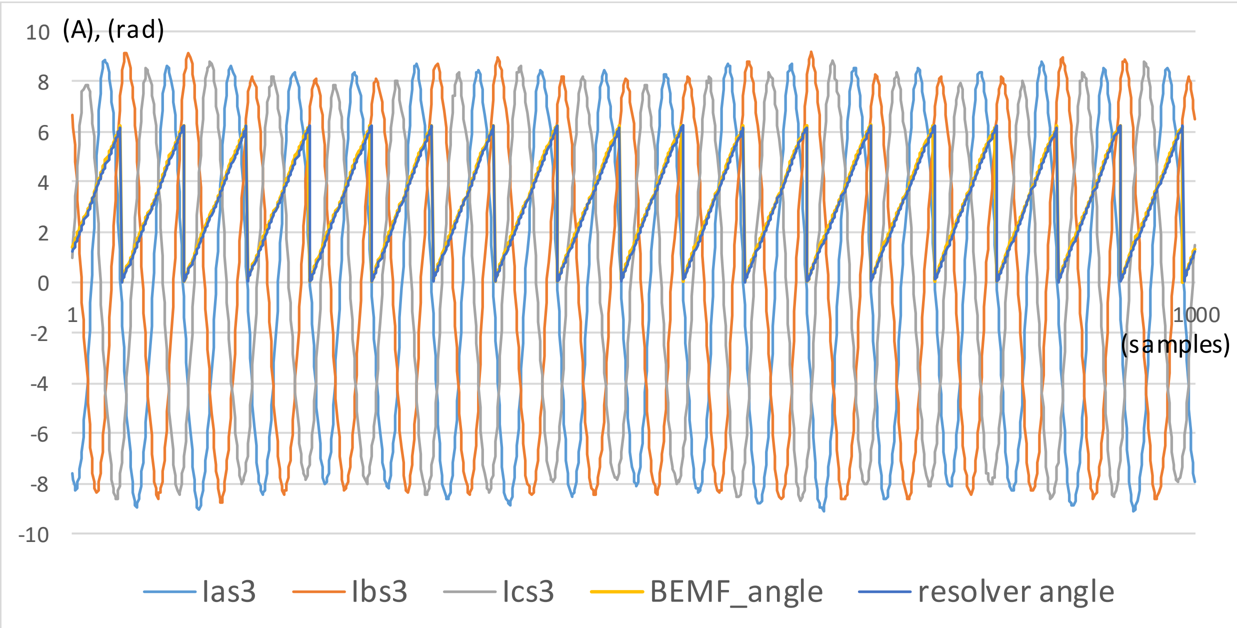


Figure 5.46 Motor spinning at 180Hz electrical (load), observer/resolver electrical angle, phase currents

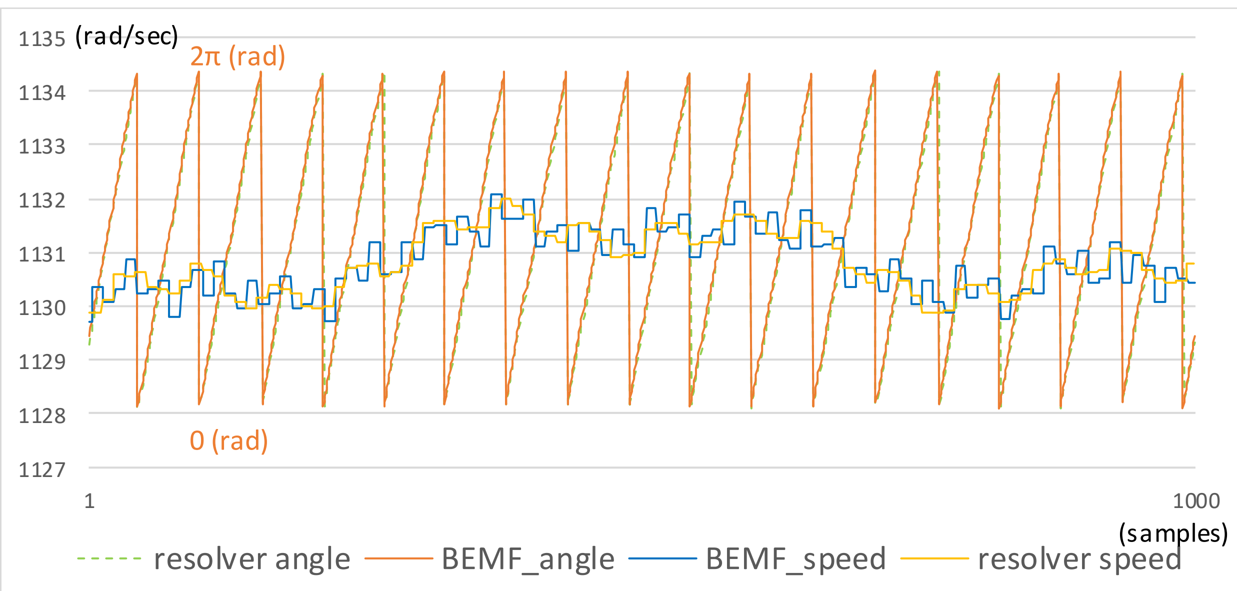


Figure 5.47 Motor spinning at 180Hz electrical (no load), observer/resolver speed (rad/sec) and electrical angle (y axis range for angle signals is 0 to 2π)

Motor spinning with no load at 200Hz (1256rad/sec) electrical, 2400rpm mechanical

Note: Sampling of variables was performed at 100 microseconds period, total of 1000 samples per variable.

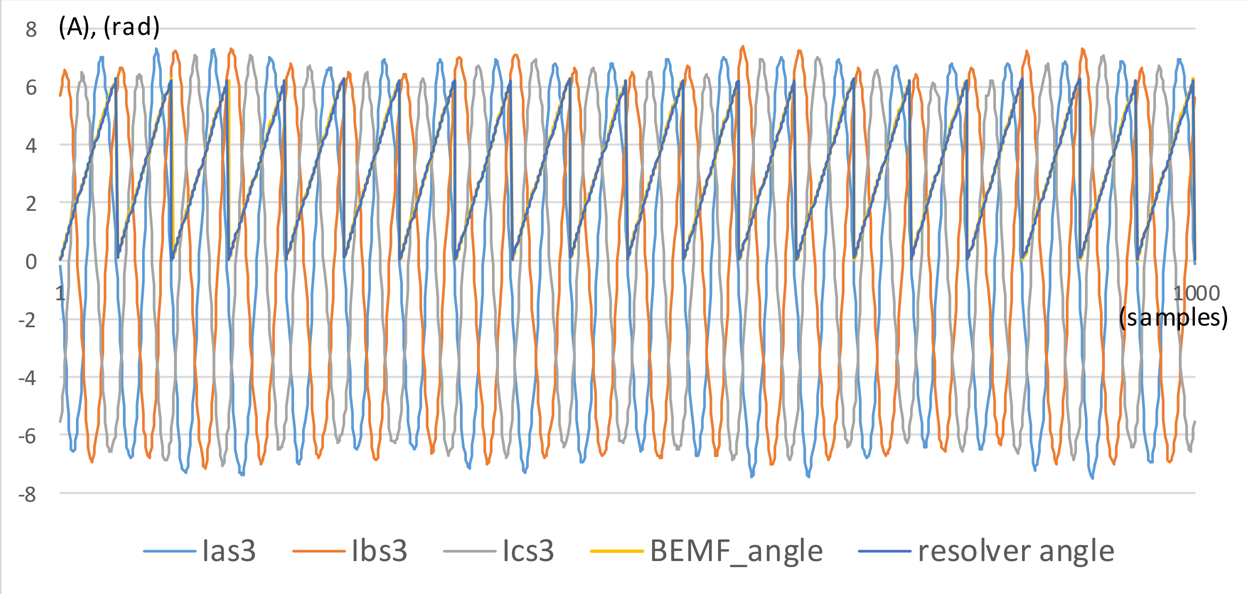


Figure 5.48 Motor spinning at 200Hz electrical (no load), observer/resolver electrical angle, phase currents

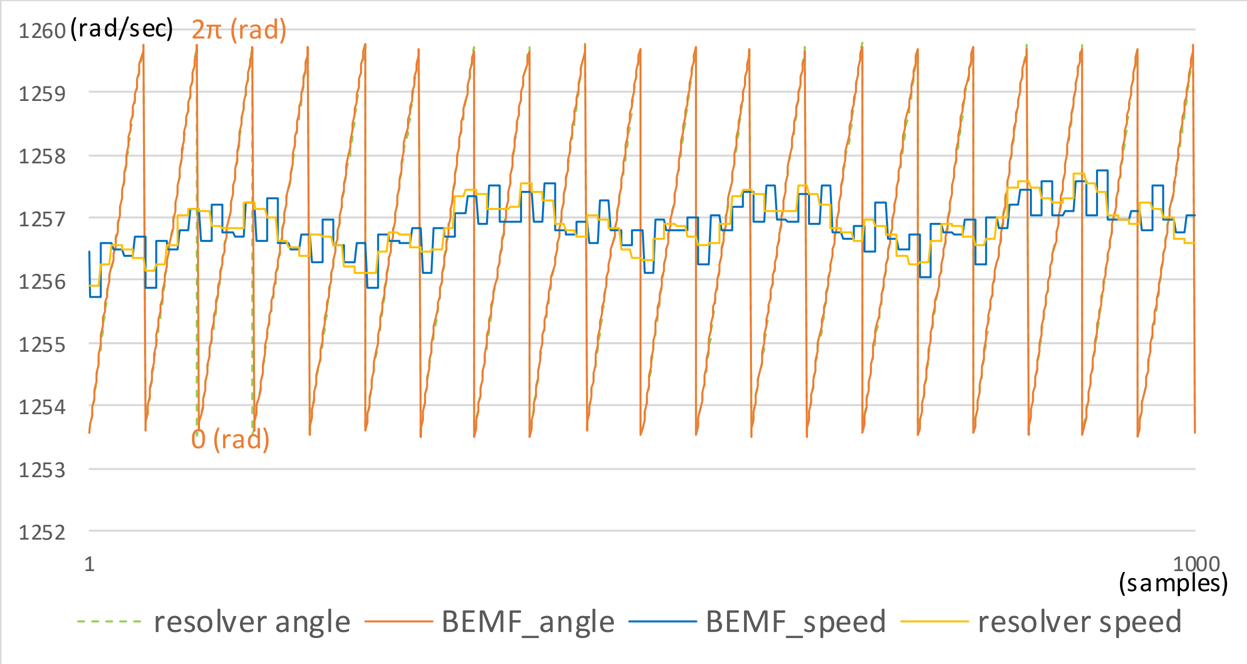


Figure 5.49 Motor spinning at 200Hz electrical (no load), observer/resolver speed (rad/sec) and electrical angle (y axis range for angle signals is 0 to 2π)

Motor spinning under load at 200Hz (1256rad/sec) electrical, 2400rpm mechanical

Note1: Sampling of variables was performed at 100 microseconds period, total of 1000 samples per variable.

Note2: A magnetic particle brake from Placid Industries (model PFB-400) was used to apply the load to the motor while spinning setting the voltage to 1V resulting to torque of ≈ 4 Nm.

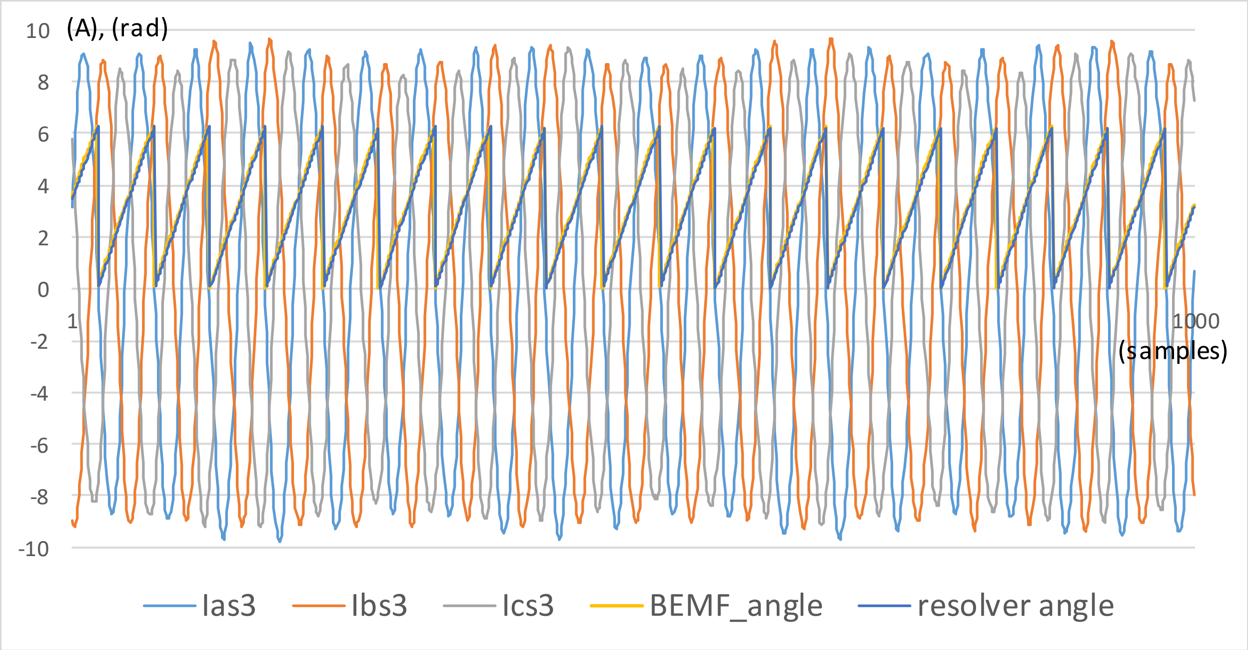


Figure 5.50 Motor spinning at 200Hz electrical (load), observer/resolver electrical angle, phase currents

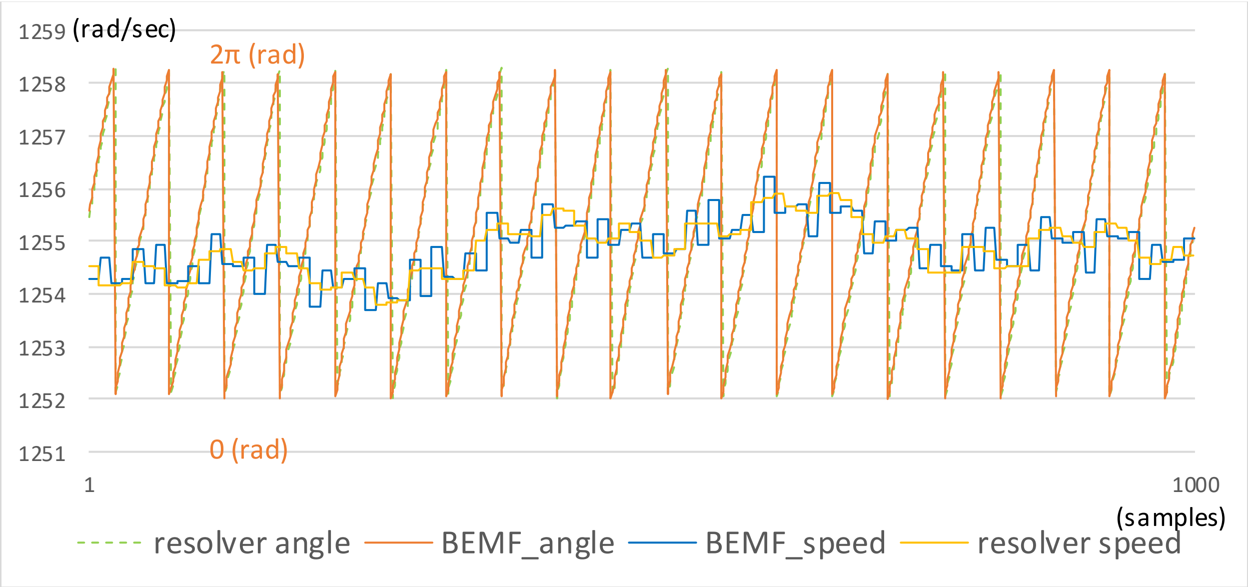


Figure 5.51 Motor spinning at 200Hz electrical (load), observer/resolver speed (rad/sec) and electrical angle (y axis range for angle signals is 0 to 2π)

Motor spinning with no load at 210Hz (1319rad/sec) electrical, 2520rpm mechanical

Note: Sampling of variables was performed at 100 microseconds period, total of 1000 samples per variable.

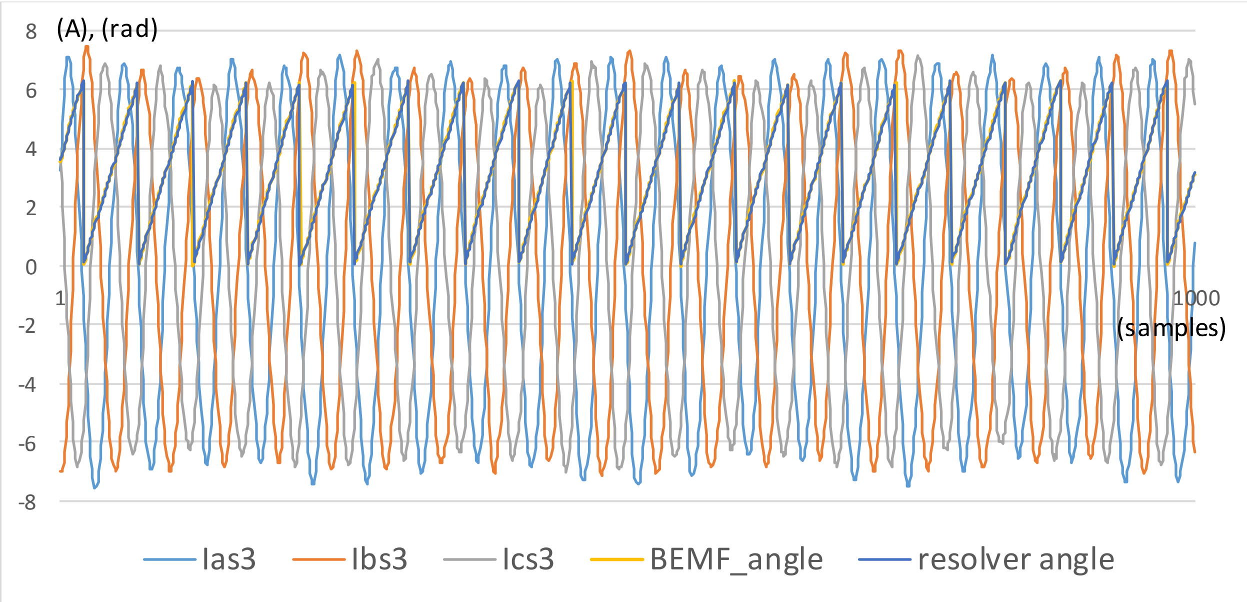


Figure 5.52 Motor spinning at 210Hz electrical (no load), observer/resolver electrical angle, phase currents

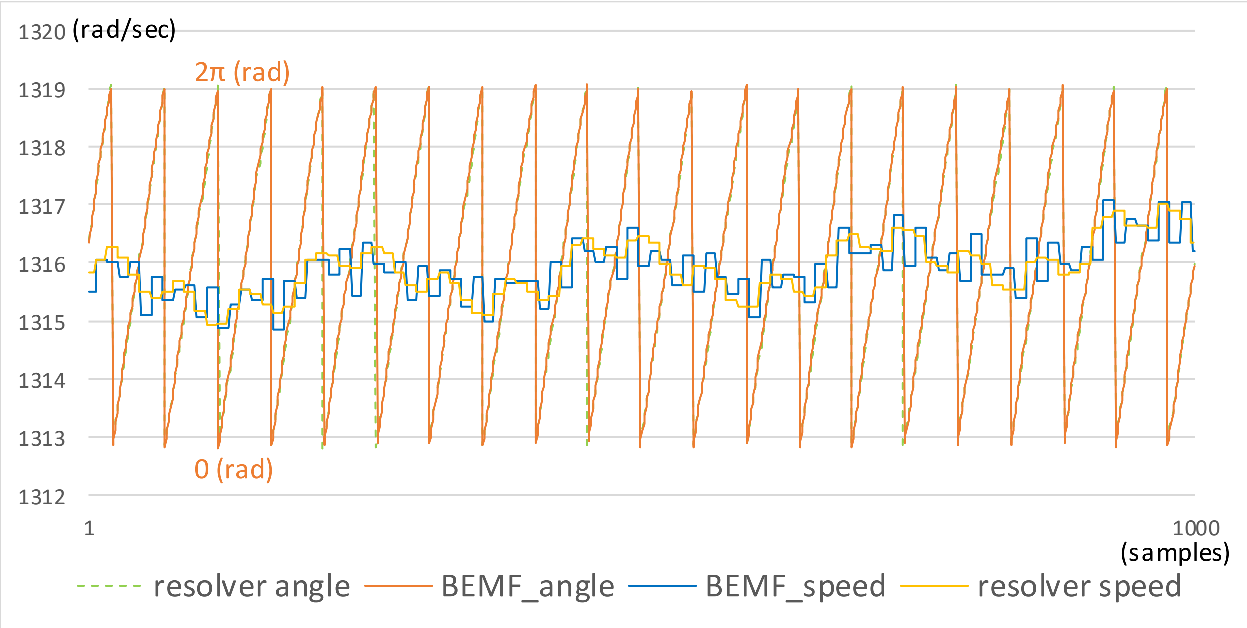


Figure 5.53 Motor spinning at 200Hz electrical (no load), observer/resolver speed (rad/sec) and electrical angle (y axis range for angle signals is 0 to 2π)

Motor spinning under load at 210Hz (1319rad/sec) electrical, 2520rpm mechanical

Note1: Sampling of variables was performed at 100 microseconds period, total of 1000 samples per variable.

Note2: A magnetic particle brake from Placid Industries (model PFB-400) was used to apply the load to the motor while spinning setting the voltage to 1V resulting to torque of ≈ 4 Nm.

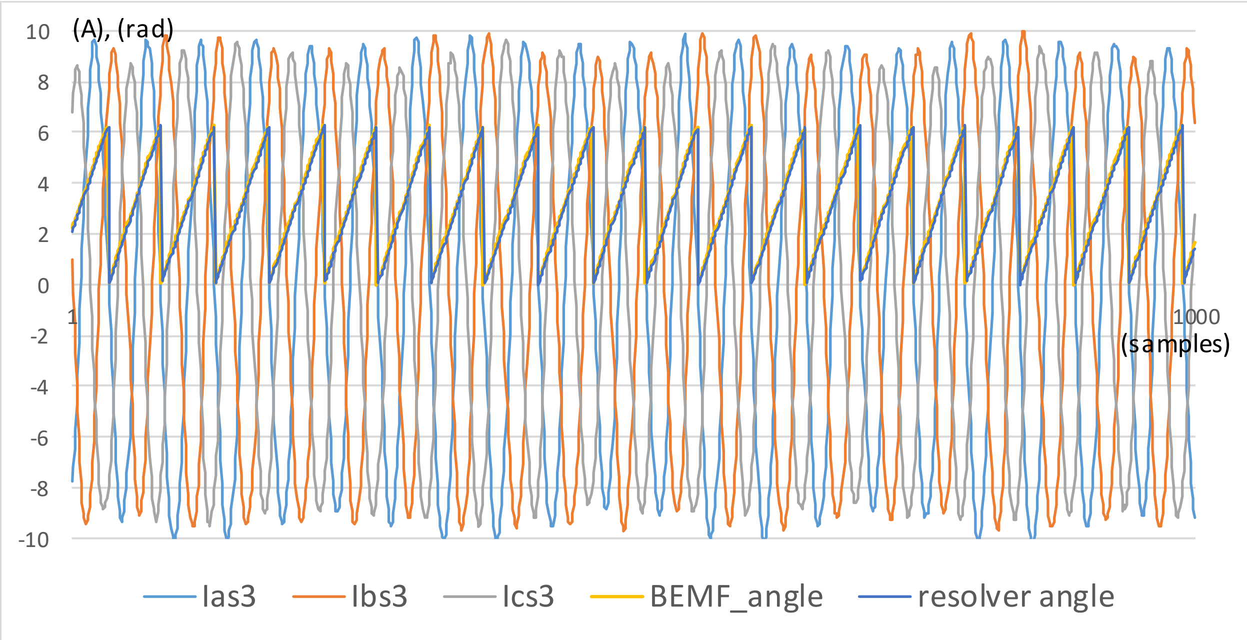


Figure 5.54 Motor spinning at 210Hz electrical (load), observer/resolver electrical angle, phase currents

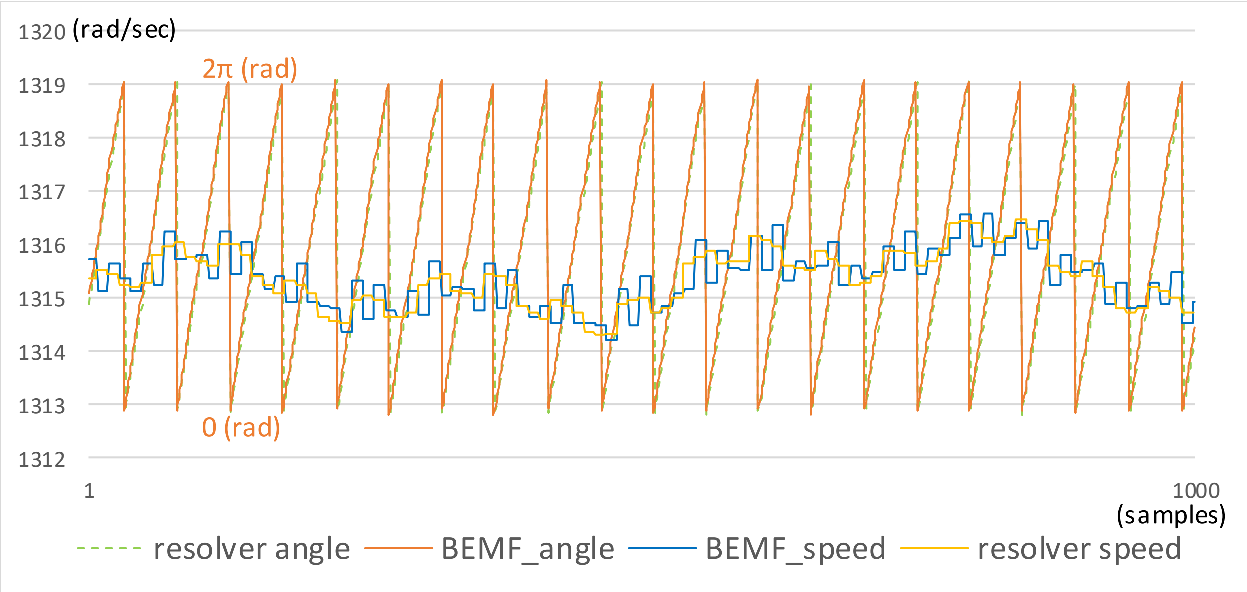


Figure 5.55 Motor spinning at 200Hz electrical (load), observer/resolver speed (rad/sec) and electrical angle (y axis range for angle signals is 0 to 2π)

Analysis of test results 30Hz to 210Hz electrical and accuracy metrics:

1. Noticing test results throughout the speed range from 360 rpm to 2520 rpm, the back EMF observer was able to track the motor's electrical angle and mechanical speed as per Figures 5.11 - 5.55. Some harmonics are observed that are analysed in points 2 and 3 below.

2. The observer's angle suffers from harmonics at 6 times the fundamental electrical frequency as due to the phenomenon described above as Dead Time Distortion (DTD).

3. There is a number of ways to mitigate the effects of DTD and specifically the $5^{th}/7^{th}$ harmonic contamination to the sensorless observer. Firstly using Dead Time Compensation (DTC) and secondly by introducing a band stop filter reducing the $5^{th}/7^{th}$ harmonics of DTD. As these two types of compensations are not the primary topic of this research it in be included in the future work section of this thesis.

Below data illustrate accuracy metrics with respect to the observer's estimated angle / speed with no load at under load. Initial analysis shows angle accuracy of about 2.1% to 6.1%. At first inspection it might appear as low accuracy, however observing Figure 5.12 shows that the majority of this error is due to the DTD $5^{th}/7^{th}$ harmonic which can be addressed and is added to future work of this thesis. Similarly the speed error ranges from 0.04% to 0.97% and once again looking at Figure 5.12 the majority of this noise is due to the DTD $5^{th}/7^{th}$ harmonic.

Motor spinning without load at 30Hz electrical (360 rpm)

Sensorless performance value evaluated		%
angle error (degrees electrical)	17.573	4.881
speed error (rpm)	3.302	0.918
peak motor phase currents (A)	5.511	N/A

Motor spinning under load at 30Hz electrical (360 rpm)

Sensorless performance value evaluated		%
angle error (degrees electrical)	16.831	4.675
speed error (rpm)	3.519	0.974
peak motor phase currents (A)	7.681	N/A

Motor spinning without load at 40Hz electrical (480 rpm)

Sensorless performance value evaluated		%
angle error (degrees electrical)	13.999	3.888
speed error (rpm)	3.208	0.665
peak motor phase currents (A)	5.925	N/A

Motor spinning under load at 40Hz electrical (480 rpm)

Sensorless performance value evaluated		**%**
angle error (degrees electrical)	17.046	4.735
speed error (rpm)	2.845	0.591
peak motor phase currents (A)	8.422	N/A

Motor spinning without load at 60Hz electrical (720 rpm)

Sensorless performance value evaluated		**%**
angle error (degrees electrical)	12.986	3.607
speed error (rpm)	2.836	0.392
peak motor phase currents (A)	6.174	N/A

Motor spinning under load at 60Hz electrical (720 rpm)

Sensorless performance value evaluated		**%**
angle error (degrees electrical)	20.574	5.715
speed error (rpm)	2.872	0.398
peak motor phase currents (A)	9.634	N/A

Motor spinning without load at 80Hz electrical (960 rpm)

Sensorless performance value evaluated		**%**
angle error (degrees electrical)	13.535	3.759
speed error (rpm)	2.511	0.261
peak motor phase currents (A)	6.346	N/A

Motor spinning under load at 80Hz electrical (960 rpm)

Sensorless performance value evaluated		**%**
angle error (degrees electrical)	22.155	6.154
speed error (rpm)	2.610	0.271
peak motor phase currents (A)	9.756	N/A

Motor spinning without load at 100Hz electrical (1200 rpm)

Sensorless performance value evaluated		**%**
angle error (degrees electrical)	9.302	2.584
speed error (rpm)	1.734	0.144

peak motor phase currents (A)	6.263	N/A

Motor spinning under load at 100Hz electrical (1200 rpm)

Sensorless performance value evaluated		**%**
angle error (degrees electrical)	17.481	4.855
speed error (rpm)	1.304	0.108
peak motor phase currents (A)	9.578	N/A

Motor spinning without load at 120Hz electrical (1440 rpm)

Sensorless performance value evaluated		**%**
angle error (degrees electrical)	9.486	2.635
speed error (rpm)	1.419	0.098
peak motor phase currents (A)	6.645	N/A

Motor spinning under load at 120Hz electrical (1440 rpm)

Sensorless performance value evaluated		**%**
angle error (degrees electrical)	16.474	4.576
speed error (rpm)	1.466	0.101
peak motor phase currents (A)	9.726	N/A

Motor spinning without load at 140Hz electrical (1680 rpm)

Sensorless performance value evaluated		**%**
angle error (degrees electrical)	9.098	2.527
speed error (rpm)	1.441	0.085
peak motor phase currents (A)	6.957	N/A

Motor spinning under load at 140Hz electrical (1680 rpm)

Sensorless performance value evaluated		**%**
angle error (degrees electrical)	14.787	4.107
speed error (rpm)	1.260	0.075
peak motor phase currents (A)	9.664	N/A

Motor spinning without load at 160Hz electrical (1920 rpm)

Sensorless performance value evaluated		**%**
angle error (degrees electrical)	8.063	2.239

speed error (rpm)	1.126	0.058
peak motor phase currents (A)	7.090	N/A

Motor spinning under load at 160Hz electrical (1920 rpm)

Sensorless performance value evaluated		**%**
angle error (degrees electrical)	12.375	3.437
speed error (rpm)	1.356	0.070
peak motor phase currents (A)	8.821	N/A

Motor spinning without load at 180Hz electrical (2160 rpm)

Sensorless performance value evaluated		**%**
angle error (degrees electrical)	7.825	2.173
speed error (rpm)	1.298	0.060
peak motor phase currents (A)	7.199	N/A

Motor spinning under load at 180Hz electrical (2160 rpm)

Sensorless performance value evaluated		**%**
angle error (degrees electrical)	12.620	3.505
speed error (rpm)	1.184	0.054
peak motor phase currents (A)	9.188	N/A

Motor spinning without load at 200Hz electrical (2400 rpm)

Sensorless performance value evaluated		**%**
angle error (degrees electrical)	7.769	2.158
speed error (rpm)	0.974	0.040
peak motor phase currents (A)	7.503	N/A

Motor spinning under load at 200Hz electrical (2400 rpm)

Sensorless performance value evaluated		**%**
angle error (degrees electrical)	13.886	3.857
speed error (rpm)	1.107	0.046
peak motor phase currents (A)	9.781	N/A

Motor spinning without load at 210Hz electrical (2520 rpm)

Sensorless performance value evaluated		**%**
angle error (degrees electrical)	7.704	2.140
speed error (rpm)	1.222	0.048
peak motor phase currents (A)	7.558	N/A

Motor spinning under load at 210Hz electrical (2520 rpm)

Sensorless performance value evaluated		**%**
angle error (degrees electrical)	14.070	3.908
speed error (rpm)	1.145	0.045
peak motor phase currents (A)	10.202	N/A

The sensorless angular and speed error data presented above in table format, are illustrated graphically in Figures 5.56 and 5.57 below.

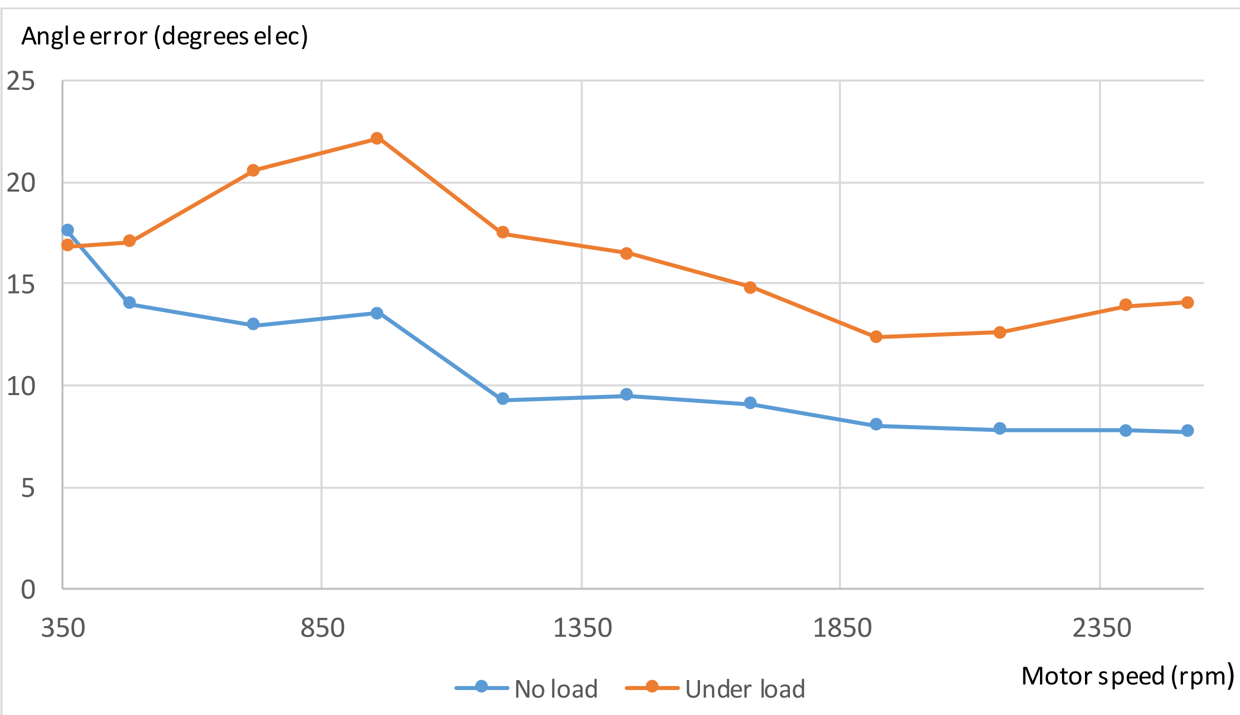


Figure 5.56 Observer angle error summary at 360-2520rpm speed range without and under load

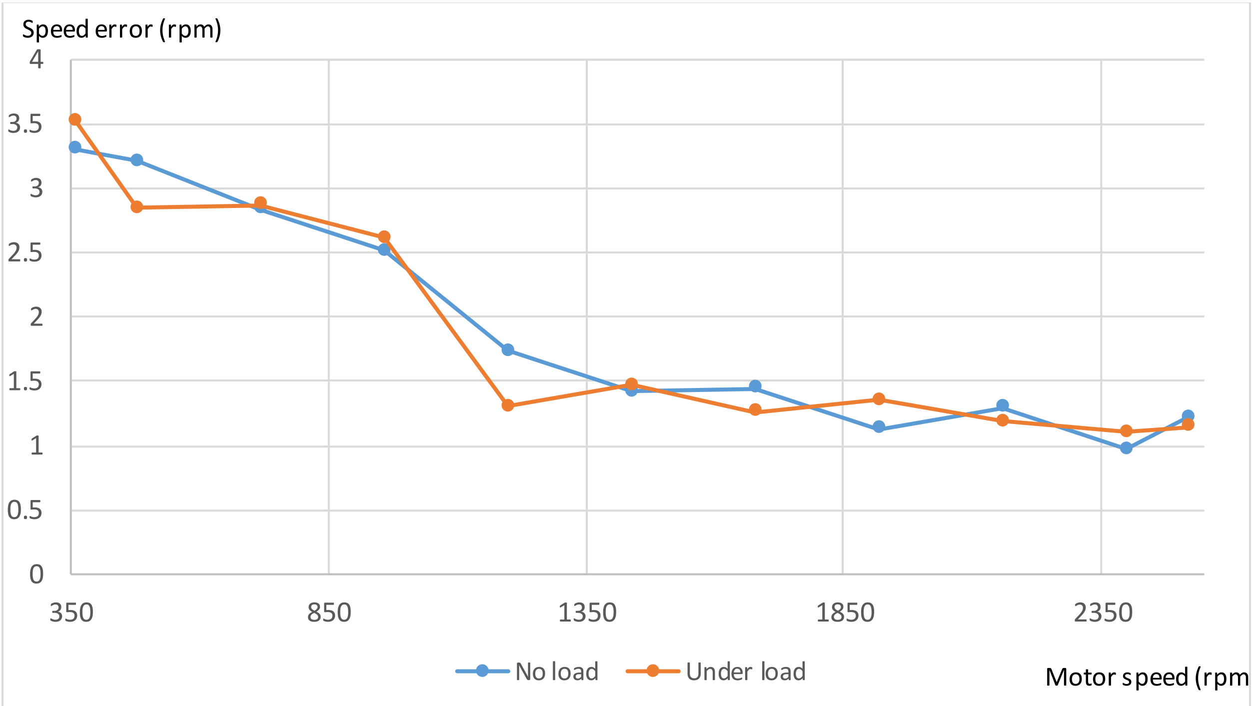


Figure 5.57 Observer speed error summary at 360-2520rpm speed range without and under load

The angle and speed accuracy tables and figures above illustrate:

1. the angle error of the observer tends to be worse under load. This is due to the controller driving harder, using higher current levels and torque, causing more sudden changes in the motor angle and the observer taking time to track these sudden changes therefore increasing the error in the estimated angle.

2. The speed error improves as the speed increases. This is because both the motor as well as the observer internal blocks behave as Low Pass Filters. As the speed increases, the higher harmonics due to DTD are attenuated by these LPFs. This is more clear in Figure 5.58 that compares the $5^{th}/7^{th}$ harmonic levels at the motor phase currents and estimated angle at two different rotational speeds (30Hz and 210Hz electrical). It can be seen that the levels of $5^{th}/7^{th}$ harmonics in the motor phase currents, in the estimated angle and as a result in the estimated speed are higher at 30Hz than at 210Hz electrical speed. This harmonic level reduction, means that both the motor and the design components (Low Pass Filters) within the observer create an attenuation of these harmonics.

3. At first sight angle error of 22 degrees may seem exceptionally high. However looking at Figures 5.11 to 5.54 the dominant part of this error is an oscillation around the target angle at $5^{th}/7^{th}$ harmonic due to DTD and a Band Pass Filter suppressing this could improve this substantially.

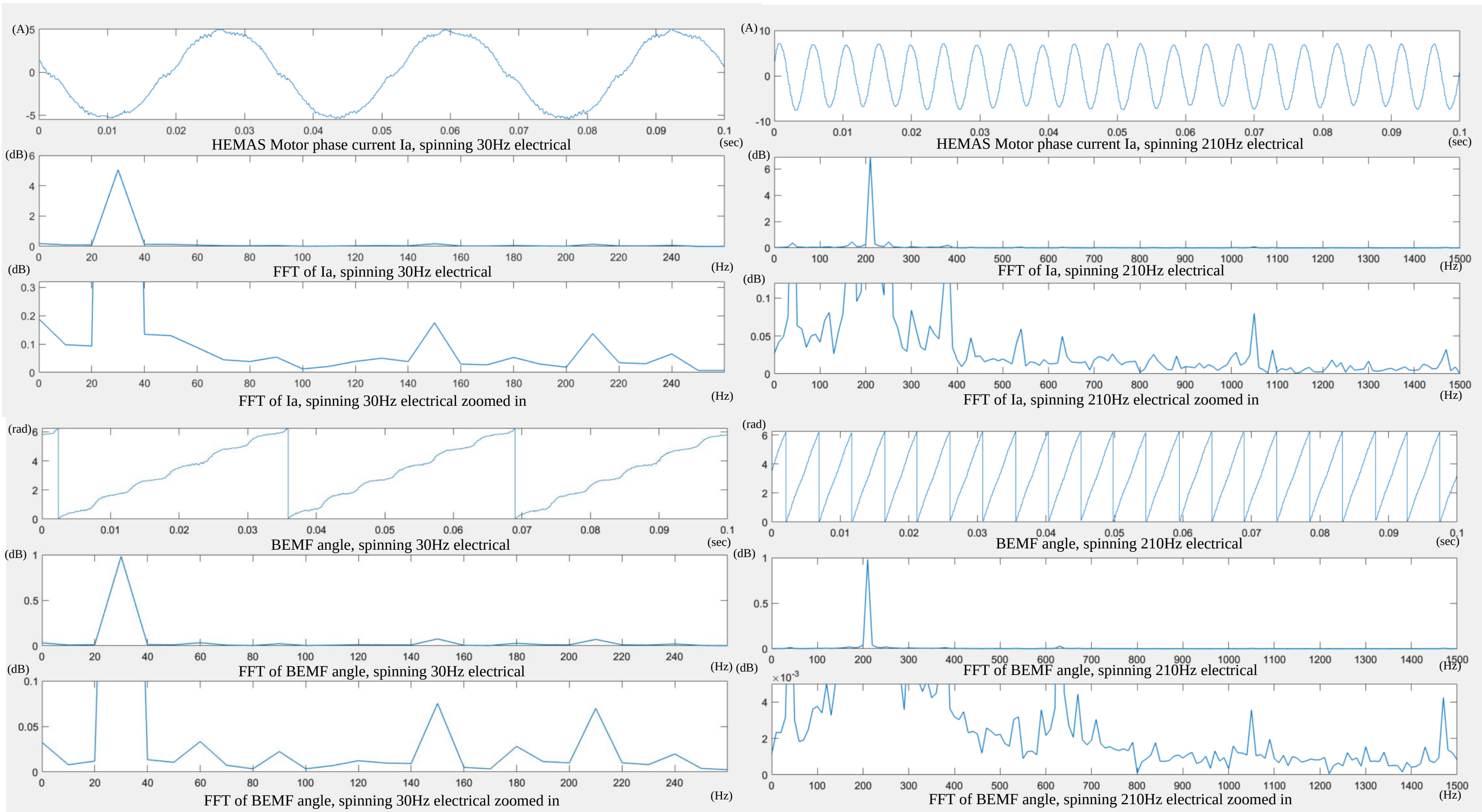


Figure 5.58 FFT analysis of motor phase current, and BEMF angle 30Hz to 210Hz electrical speed

Dynamic testing during motor acceleration

Note: Sampling of variables was performed at 1ms period, total of 1000 samples per variable.

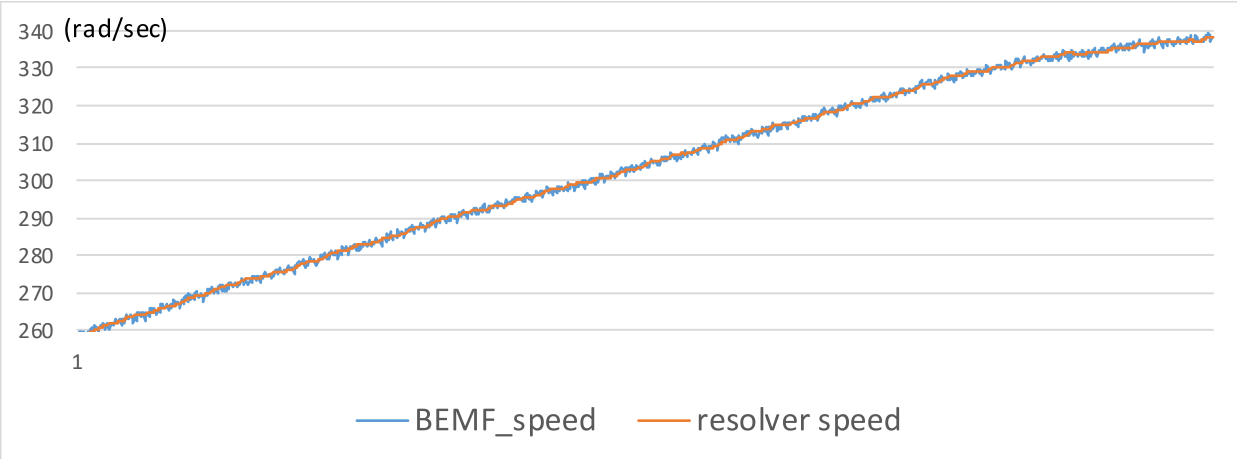


Figure 5.59 Motor accelerating, observer/resolver electrical speed (rad/sec)

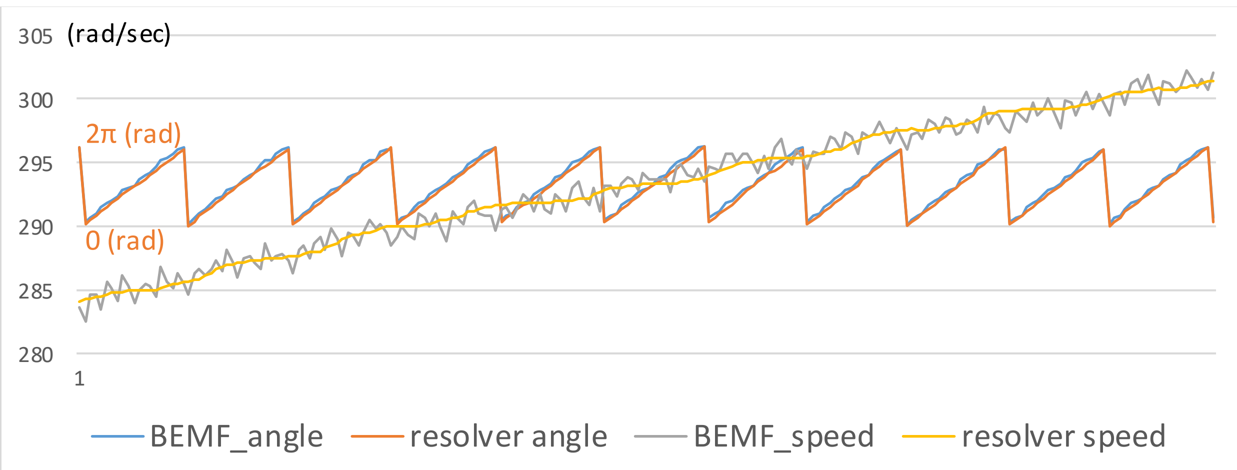


Figure 5.60 Motor accelerating, observer/resolver electrical speed (rad/sec) and angle zoomed in

Analysis of test results while motor is accelerating:

1. The observer is able to track the motor's electrical angle and speed with the limitation of the $5^{th}/7^{th}$ harmonic content that can be eliminated as described above.

2. To enable the tracking of the motor values during acceleration the sampling period was lowered 10 times versus above tests from 100 microseconds to 1ms, so as to capture a larger time window. This reduction in sampling has an impact on plotting detail, however the main observations from previous tests are still apparent.

3. Note that in figure 5.60 the unit in y axis is rad/sec and intended for the speed, the units for the angle are not displayed in this graph but range is from 0 to 2π.

5.8 Overall analysis of simulation and test results

Matlab simulation presented in section 5.2 (Figures 5.2/5.3) enabled high level understanding of the back EMF voltage characteristics for the HEMAS motor. Transitioning to more detailed clock accurate VHDL simulation in section 5.5, lowered the algorithm to implementation level HDL and evaluated various consideration incorporating motor and inverter models along with PWM switching. The confidence obtained by means of simulation, was vital towards the implementation in C for the HEMAS platform. The observer was tested from medium to maximum speed supported by the test rig and at various load conditions. During hardware testing DTD introduced a $5^{th}/7^{th}$ harmonic to the observer's angle and speed. This effect was not modelled and therefore was not identified during simulation. However, these harmonics can be removed either by DTC or with the introduction of a dynamic bandstop filter within the observer. The observer worked well on hardware tracking the motor position and speed throughout the speed range and during acceleration.

In terms of simulation/testing results comparison, while simulation was absolutely vital and essential towards preparing for the implementation, the DTD phenomenon did create differences between simulation and testing which were understood and analysed in section 5.7.

5.9 Conclusions

The chapter intended to evaluate a model based observer in terms of simulation and hardware testing on the HEMAS platform. The observer is proposed to be used towards increasing system availability in case of a resolver failure. The chapter first introduced the theoretical background of the observer including the PMSM's equivalent circuit and mathematical formulas. Matlab and VHDL Modelsim simulation results were then presented targeting the HEMAS motor. The implementation steps of the observer were then illustrated, followed by hardware testing results on the HEMAS test rig covering different speed and load conditions. Experimental testing showed that the sensorless observer can track the rotor velocity and angle over a wide speed range.

Chapter 6 Saliency Based Method

6.1 The concept of saliency

Permanent Magnet Synchronous Motors (PMSMs) can be classified into two types, salient and non-salient [8]. This classification relates to the rotor's magnetic field that is in turn dependent on the location of the permanent magnet within the rotor. IPMSMs are therefore considered salient by design (geometric saliency) with projecting poles and $L_d \neq L_q$ where L_d, L_q is the motor inductance as observed in the direct and quadrature synchronous axis. On the other hand SPMSMs tend to be mostly non salient where L_d is almost equal to L_q. Although SPMSMs are non-salient by design they can exhibit mild saliency when sufficient current is placed on the d axis causing stator iron saturation [72], [73]. It is this saliency by design in IPMSMs [74], [75] and by stator iron saturation in SPMSMs [72], [73] that is used to establish saliency based sensorless methods investigated in this chapter.

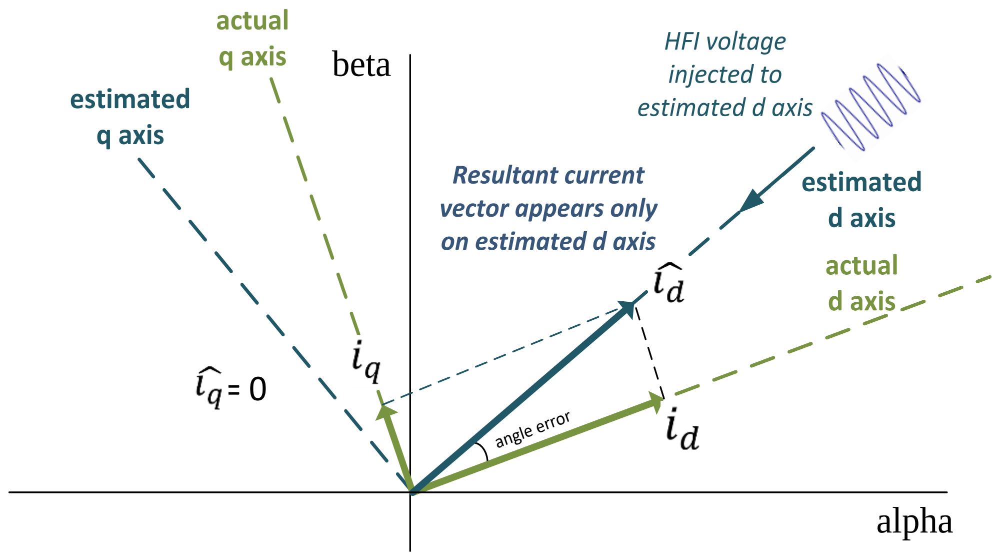
The HEMAS motor is a Surface Mounted Permanent Magnet Synchronous Motor (SPMSM) and therefore as $Ld \approx Lq$ (see table 6.1) stator iron saturation needs to be introduced to establish saliency. This chapter provides the theory, simulation and test results for a saliency based sensorless method applied to the HEMAS system.

6.2 The concept of saliency

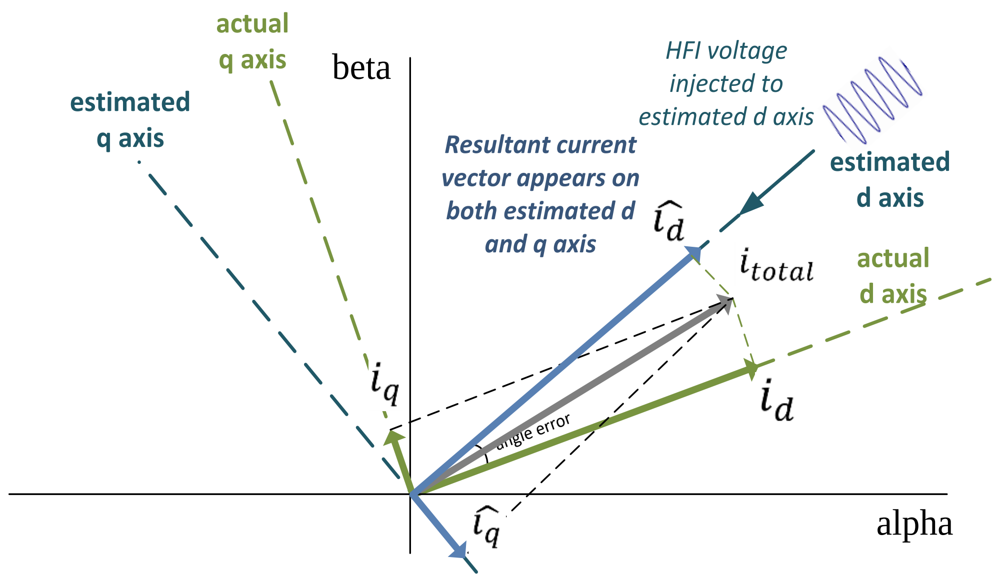
HFI methods are based on the concept that if $Ld \neq Lq$, the controller can assume an estimated rotor angle $\hat{\vartheta}$ and then inject a HF voltage component in the stationary or synchronous frames. If there is an error in the estimated angle, this error is observable in the HF feedback currents. Specifically as $Ld \neq Lq$, if the injected voltage is placed on the d axis that generally has lower inductance it produces HF currents of higher amplitude relative to the q axis adhering to the equation below derived by Faraday's law:

$$V = L\frac{dI}{dt} \tag{Eq 6.1}$$

Another way to visualise the basic principle of saliency in sensorless methods is shown in Figure 6.1 and analysed below. Consider that $L_d = L_q$ and the motor controller assumes an estimated rotor angle $\hat{\vartheta}$. If a High Frequency voltage is placed to the estimated d axis of the motor, this voltage will be naturally decomposed to the actual d and q axis of the motor as a function of the angle error $\Delta\vartheta$ ($V_d = V_{HFI} * \cos \Delta\vartheta$ and $V_q = V_{HFI} * \sin \Delta\vartheta$, where ϑ is the rotor electrical angle, $\hat{\vartheta}$ is the estimated angle by the sensorless method and $\Delta\vartheta = \vartheta - \hat{\vartheta}$ is the angular error). Irrespective of the angle error $\Delta\vartheta$ and the subsequent d/q axis voltage sharing, as $L_d = L_q$ the resultant vectorial summation of the d and q currents will always add to the same vector as each axis responds equally to a given voltage (see Figure 6.1.a). An external observer that would therefore place a HF voltage to the estimated d axis would always observe the resultant current on its estimated d axis current $\hat{i_d}$ (Figure 6.1.a). However if $L_d \neq L_q$ and a HF voltage is applied to the estimated d axis with an angular error $\Delta\vartheta$, the voltage deposited on the axis with lower inductance will create higher amplitude HF currents. This imbalanced d/q current response to the HF voltage causes the vectorial addition of the d/q axis currents to be misaligned versus the estimated d axis (see Figure 6.1.b). The external observer would therefore notice that although the HF voltage is deposited on the estimated d axis, HF current is seen on both the estimated d and q axis as a function of the angle error (Figure 6.1.b).



6.1.a Ld = Lq



6.1.b Ld ≠ Lq

Figure 6.1 HFI and effect of saliency in feedback currents

Note 1: signals in this thesis containing the hat accent ^ for example $\hat{\vartheta}, \widehat{V_d}, \widehat{V_q}, \widehat{\iota_d}, \widehat{\iota_q}$ are values estimated by the controller, i.e. the estimated rotor angle and corresponding voltages/currents to the estimated synchronous frame.

Note 2: the HFI injection frequency is chosen to be higher than the mechanical time constants of the motor drive typically >= 1kHz so as to minimize the effect on observed torque ripple and also avoid interaction with the current control loop.

Note3: HFI can be applied in the form of sinusoidal or square wave and to either the stationary or synchronous frame.

For sinusoidal High Frequency Injection applied to the stationary frame also known as rotating injection equations 6.2 and 6.3 apply:

$$\hat{V}\alpha_{injected} = V_{carrier} * (\sin{(\omega_c t)}) \qquad \text{Eq 6.2}$$

$$\hat{V}\beta_{injected} = V_{carrier} * (\cos{(\omega_c t)}) \qquad \text{Eq 6.3}$$

Sinusoidal injection to the synchronous frame also known as pulsating injection involves superimposing a HF sinusoid to the d axis voltage demand. Different flavours of this injection type exist in publications where either $(sin\omega_c t)$, $(-sin\omega_c t)$ or $(cos\omega_c t)$ is injected to the d axis. However these variations are effectively equivalent with respect to their function i.e. injecting a sinusoidal voltage to the d axis, and observing the high frequency feedback current components and primarily differ in the mathematical details of their equivalent formulas. The injection convention that is to be used in this research is $(-sin\omega_c t)$ as per equation 6.4:

$$\hat{V}d_{injected} = V_{carrier} * (-sin(\omega_c t)) \qquad \text{Eq 6.4}$$

6.3 Theory of HFI and the proposed demodulation method

While the pulsating injection signal waveform investigated in this chapter is a sinusoid of constant frequency, chapter 7 presents a method where the injection frequency and waveform shape vary for acoustic improvement purposes.

A starting point towards understanding the effect of HFI is revisiting the equivalent mathematical model of a PMSM in the synchronous frame [63]:

$$\begin{bmatrix} V_d \\ V_q \end{bmatrix} = \begin{bmatrix} R_s i_d + \frac{d}{dt}\Psi_d - \omega_r \Psi_q \\ R_s i_q + \frac{d}{dt}\Psi_q + \omega_r \Psi_d \end{bmatrix} \qquad \text{Eq 6.5}$$

where:

$$\Psi_d = L_d i_d + \Psi_f \qquad \text{Eq 6.6}$$

$$\Psi_q = L_q i_q \qquad \text{Eq 6.7}$$

V_d V_q are stator voltages in the synchronous frame, R_s is the stator resistance, i_d and i_q are stator currents in d and q axis, ω_r is the rotor electrical angular speed, L_d is the d axis inductance, L_q is the q axis inductance, and Ψ_f is the rotor flux.

If the motor is spinning at low frequency the back EMF voltage that is proportional to the motor speed ω_r can be neglected and the motor can be approximated by a simplified R,L circuit. Equation 6.5 therefore becomes:

$$\begin{bmatrix} V_d \\ V_q \end{bmatrix} = \begin{bmatrix} R_s i_d + \frac{d}{dt}\Psi_d \\ R_s i_q + \frac{d}{dt}\Psi_q \end{bmatrix} \qquad \text{Eq 6.8}$$

Furthermore when a HFI method is applied, the voltage drop due to the motor's inductance tends to dominate over the voltage drop due to the motor's resistance. The circuit can thus be simplified further and equation 6.8 becomes:

$$\begin{bmatrix} V_d \\ V_q \end{bmatrix} = \begin{bmatrix} \frac{d}{dt}\Psi_d \\ \frac{d}{dt}\Psi_q \end{bmatrix} \qquad \text{Eq 6.9}$$

Considering equations 6.6, 6.7, and that $\frac{d}{dt}\Psi_f = 0$ equation 6.9 becomes:

$$\begin{bmatrix} V_d \\ V_q \end{bmatrix} = \begin{bmatrix} L_d \\ L_q \end{bmatrix} * \frac{d}{dt}\begin{bmatrix} i_d \\ i_q \end{bmatrix} \qquad \text{Eq 6.10}$$

Assume that the estimated rotor angle by the controller contains an error $\Delta\vartheta$ where $\Delta\vartheta = \vartheta - \hat{\vartheta}$ is the angular error, ϑ is the rotor electrical angle, $\hat{\vartheta}$ is the estimated angle by the sensorless method, $\Sigma L = (L_d + L_q)/2$ and $\Delta L = (L_q - L_d)/2$. Substituting $\widehat{V_d} = V_{carrier}(-sin(\omega_c t))$ $\widehat{V_q} = 0$. and solving for $\widehat{\iota_d}$, $\widehat{\iota_q}$ results into equation 6.11:

$$\begin{bmatrix} \widehat{\iota_d} \\ \widehat{\iota_q} \end{bmatrix} = \frac{V_{carrier}}{\omega_c(\Sigma L^2 - \Delta L^2)} \begin{bmatrix} \Sigma L + \Delta L * \cos(2\Delta\vartheta) \\ \Delta L * \sin(2\Delta\vartheta) \end{bmatrix} * \cos(\omega_c t) \qquad \text{Eq 6.11}$$

Focusing on the estimated q axis current $\widehat{\iota_q}$ of equation 6.11 results into equation 6.12:

$$\widehat{\iota_q} = \frac{V_{carrier}}{\omega_c(\Sigma L^2 - \Delta L^2)} * \Delta L * \sin(2\Delta\vartheta) * \cos(\omega_c t) \qquad \text{Eq 6.12}$$

Brief analysis of the feedback current equation 6.12:

- equation 6.12 is in conceptual alignment with the visual description of the saliency methods shown in Figure 6.1. If saliency exists within a motor i.e. $\Delta L \neq 0$ and a HF voltage is applied to the estimated d axis, the HF current that is observed on the estimated q axis $\widehat{\iota_q}$ is a function of the angular error $\Delta\vartheta$.

- The HF carrier on $\widehat{\iota_q}$ is proportional to the saliency of the motor, i.e. the higher the ΔL the higher the amplitude of the HF feedback and therefore the more accurate the tracking algorithm may be. SPSMs with low levels of saliency are therefore a bit harder to apply saliency methods.
- The HF carrier on $\widehat{\iota_q}$ is proportional to the amplitude of the voltage carrier $V_{carrier}$. In other words, for a given motor's saliency, the higher the HF voltage, the higher the amplitude of the HF feedback carrier seen in $\widehat{\iota_q}$ and therefore the more accurate the tracking algorithm may be. A compromise-based analysis is therefore needed on selecting the amplitude of the HF voltage that provides the right balance between tracking accuracy and power invested on tracking. This analysis is a bit more complex when the injected current also provides the motor saliency as is the case for SPMSMs and will be explained in section 6.4 of this thesis.
- The HF feedback current is a function of $\sin(2\Delta\vartheta)$ and not $\sin(\Delta\vartheta)$. This will be revisited below under "initial rotor position estimation algorithms".

If it was somehow possible to calculate the value of $\Delta\vartheta$ by solving equation 6.12, this could feed to a rotor angle tracking algorithm. To achieve this, a method known as direct demodulation is used that involves multiplying $\widehat{\iota_q}$ with $\cos(\omega_c t)$ as shown in equation 6.13:

$$\widehat{\iota_q} * \cos(\omega_c t) = \frac{V_{carrier}}{2 * \omega_c(\Sigma L^2 - \Delta L^2)} * \Delta L * \sin(2\Delta\vartheta) * (1 + \cos(2\omega_c t)) \qquad \text{Eq 6.13}$$

By passing the equation 6.13 through a Low Pass Filter, the $\cos(2\omega_c t)$ can be eliminated resulting into:

$$LPF(\widehat{\iota_q} * \cos(\omega_c t)) \approx \frac{\boldsymbol{V_{carrier}}}{\boldsymbol{2 * \omega_c(\Sigma L^2 - \Delta L^2)}} * \Delta L * \sin(2\Delta\vartheta) \qquad \text{Eq 6.14}$$

Note that the use of a LPF introduces some latency in the calculation of the estimated angle, placing the Cut Off Frequency (COF) very low, would filter the HF voltage well but would introduce a latency in the calculation.

As the elements of equation 6.14 in bold are constants or of known value for a specific motor and HFI algorithm, a rotor angle tracker can use the angle error ε**,** where ε:

$$\varepsilon = LPF\left(\widehat{\iota_q} * \cos(\omega_c t)\right) = hfi\_motor\_params * \sin(2\Delta\vartheta) \qquad \text{Eq 6.15}$$

where $hfi\_motor\_params = \frac{V_{carrier}}{2 * \omega_c(\Sigma L^2 - \Delta L^2)} * \Delta L$

Feeding ε of equation 6.15 to a tracker would achieve:

- If $\Delta\vartheta = 0$ $i.e.$ $\vartheta = \hat{\vartheta}$ the error ε would be zero
- If $\Delta\vartheta$ begins becoming positive i.e. ϑ slightly greater than $\hat{\vartheta}$ then ε would be positive
- If $\Delta\vartheta$ begins becoming negative i.e. ϑ slightly less than $\hat{\vartheta}$ then ε would be negative and therefore the tracker can continuously track the rotor's electrical angle.

Initial rotor position estimation algorithms: One key point to note is that as per equation 6.12 and 6.15 the saliency algorithm calculates an error that is proportional to $\sin(2\Delta\vartheta)$ and not

$\sin(\Delta\vartheta)$. This means that the error will appear 0 both when $\vartheta = \hat{\vartheta}$ as well as when $\vartheta = \hat{\vartheta} + 180^0$. In order to identify which of the two assumptions is true, initial position algorithms have been proposed in a number of publications. [74] proposes a two-step initial rotor position estimation where a voltage pulse is applied to the estimated d axis for each of the two angle assumptions sequentially. If the assumption is correct, it will introduce stator iron saturation, reduce L_d and increase the amplitude of the feedback current pulse. If the angle assumption is incorrect i.e. $\vartheta = \hat{\vartheta} + 180^0$ the stator iron will not saturate and there will be no decrease to L_d observed by stator. The algorithm proposed in [74] therefore identifies the correct angle assumption based on which voltage pulse introduced the higher amplitude feedback current pulse. An even better proposal than the one in [74] and covering both stationary and spinning motor is detailed in section 6.11 of this thesis proposing a novel saliency method.

Note that saturation rotor polarity detection is only needed to be performed once and from then onwards the angle is tracked by the demodulation algorithm. Another way of achieving initial position calculation in cases where up to one electrical rotation is acceptable is achieved by a method known as initial alignment [76]. Under this method, the voltage applied to the stator windings is such that it creates a magnetic field with poles at fixed electrical angle therefore initiating the rotor to rotate and align its N/S poles with the equivalent S/N poles of the stator's magnetic field.

Proposed demodulation steps on the HEMAS platform: below is the sequence of steps implementing the saliency method for the HEMAS platform:

- Inject a high frequency carrier is to the estimated d axis $\hat{V}d_{injected} = V_{carrier} * (-sin\omega_c t)$.
- The estimated d axis feedback current $\widehat{\iota_q}$ is passed through a High Pass Filter (HPF) aiming to remove the fundamental frequency component and only preserve the effect of the HFI.
- The high frequency component of $\widehat{\iota_q}$ is multiplied with $\cos(\omega_c t)$.
- A LPF is applied to remove the $\cos(2\omega_c t)$ component.
- The calculation from the last step is used as an error to track the rotor angle using a PI tracking loop as shown in Figure 6.2.

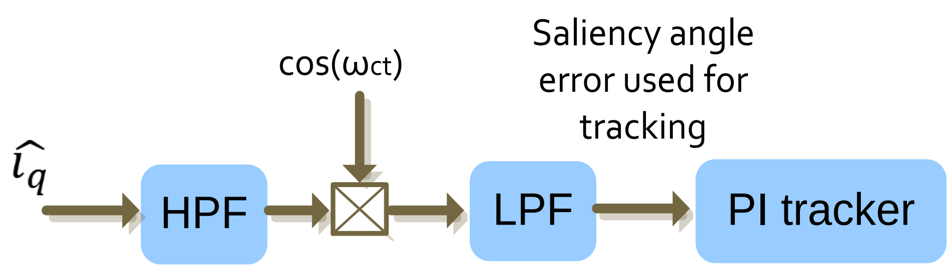


Figure 6.2 Block diagram of saliency tracking

6.4 Stator iron saturation for the HEMAS platform

6.4.1 The mechanism of iron saturation

As per equation 6.11 a motor's saliency is a key characteristic for a successfully application of saliency methods. What has not been explained though is what is the mechanism that introduces saliency for an SPMSM. Establishing high levels of flux density B i.e. creating a very strong magnetic field is a desirable feature for a magnetic circuit as this can produce high levels of torque within a motor as per equation 1.1. While theoretically it would be ideal if the stator's iron flux density B increased indefinitely, in practice its value cannot increase over 2 Tesla [9]. Attempting to do so increases what is known as the iron's magnetic reluctance and introduces a phenomenon known as iron saturation. Due to this effect, the flux density is naturally limited below this threshold as shown in Figure 6.3. Motor designs therefore tend to restrict the value of flux density between 1.6 and 1.8 Tesla [9].

Most SPMSM designs in order to obtain maximum power density are at borderline saturation point as shown in point A of Figure 6.4 where the flux density has its maximum linear point and $L_d = L_q$ [77]. However if HFI is applied to the d axis, the d axis flux density can be pushed to the non-linear region i.e. above point A of figure 6.4. In the saturated region the flux linkage (Ψ) will increase in a nonlinear manner resulting into the decrease of L_d [77]. It is this method known as saliency by means of stator iron saturation that is implemented for the HEMAS platform and presented in this chapter.

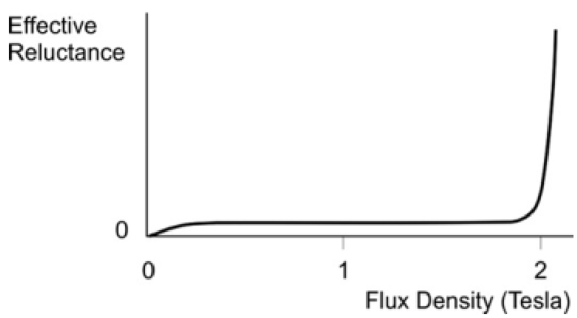


Figure 6.3 stator iron saturation

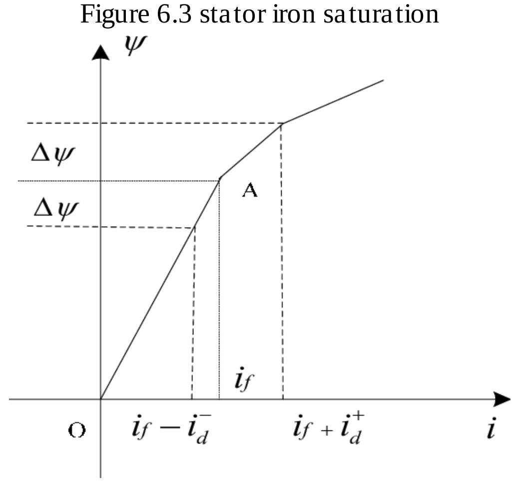


Figure 6.4 typical d axis flux linkage characteristic curve for SPMSM

6.4.2 HEMAS saturation point selection

One key step towards establishing saliency to the HEMAS motor is identifying the motor current that needs to be placed on the d axis to introduce stator iron saturation. In order to identify this point the HEMAS saliency table (Table 6.1) was used also depicted graphically in Figure 6.5. Note that this saliency table was obtained by importing the motor design in Motor-CAD. Analyzing Table 6.1 and Figure 6.5 one possible d axis current point is 5.21A providing maximum saliency of $\frac{L_q}{L_d} = 1.083$. For i_d currents greater than 5.21A the saliency in fact decreases so investing more power would not produce any useful rewards in terms of saliency. The motor parameters presented in Table 4.1 are also listed in Table 6.1 below.

Parameter	Value
No. of pole pairs p	5
Phase Resistance R_S	0.23 [Ω]
Phase Inductance L	1.193 [mH]
Voltage constant κ_E	0.092 [V/rads]
Torque constant κ_T	0.142 [Nm/A]

Table 6.1. HEMAS motor parameters when not in saturation

d axis current, positive	d axis inductance Ld (mH)	q axis inductance Lq (mH)	Lq/Ld
10.26	1.055	1.133	1.074
7.76	1.064	1.145	1.076
5.21	1.069	1.158	1.083
2.61	1.136	1.185	1.043
0	1.193	1.194	1.001

Table 6.2 HEMAS motor saliency table

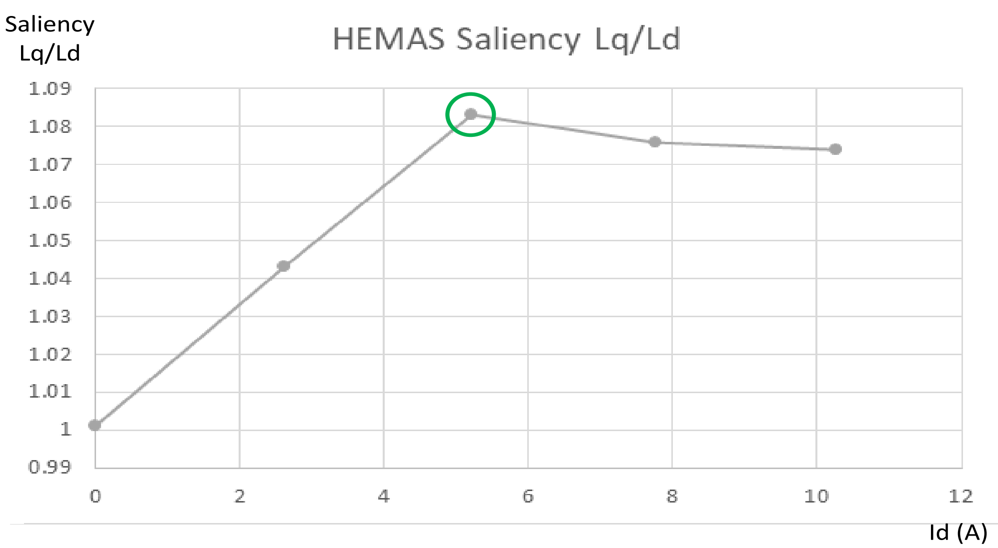


Figure 6.5 HEMAS motor saliency graph

Note that skin effect and Eddy currents are not expected to have an effect on the introduced saliency as the resulting effect will impact both d and q axis and will not affect the induced saliency due to saturation [78].

One key point to note analyzing the above table and figure is that SPSMs and the HEMAS motor exhibit relatively low levels of saliency. An example saliency ratio for an IPMSM is as per [79] $L_q/L_{d=}1.92$ versus HEMAS' $L_q/L_{d=}1.083$. This is of interest because based on equation 6.15 the signal used to track the electrical angle for a given HFI method is proportional to the formula $\frac{\Delta L}{(\Sigma L^2 - \Delta L^2)}$. Substituting ΔL and ΣL with the equivalent values of the HEMAS' and the IPMSM's of [79], this tracking signal is calculated to be 2.6 times greater for the [79] versus the HEMAS motor. Added to this, the maximum HEMAS inductance ratio $L_q/L_{d=}1.083$ is the saliency observed at 0 angular error $\Delta\vartheta$ when all of the total HFI current of 5.21 A is placed on the motor's d axis. As the angular error increases, less current is placed on the d axis resulting into the reduction of saliency and therefore an additional decrease in the amplitude of the saliency tracking signal. This saliency reduction due to angular error as will be demonstrated in the simulation and test sections results into a further reduction of the tracking signal amplitude by a factor of 3.5 for the HEMAS motor. So eventually the tracking saliency signal for the HEMAS motor is 2.6*3.5 = 9.1 times smaller than the IPMSM of [79]. The above factors make saliency algorithms for SPMSMs harder to implement relative to IPMSMs. Despite this application specific difficulty, the superior power density of SPMSMs and the advantages listed in Chapter 1 often make them the best possible choice in aerospace (see Tables 1.1 and 1.2), thus this investigation of saliency algorithms on the HEMAS motor.

6.5 Simulation/Test plan for the HEMAS saliency method

Aiming to demonstrate the proposed saliency method for the HEMAS motor, a sequence of simulation and test steps are proposed in the below plan. Matlab modelling is intended to be the first stage of simulation. The Matlab environment is expected to serve as an excellent tool to quickly analyse the anticipated current feedback signals from HFI as well as the demodulation algorithm based on theoretical formulas. This simulation is proposed to take place at high level of abstraction without attempting to simulate implementation considerations such as data sampling, PWM switching, and motor models.

VHDL modelling is proposed as the second stage of simulation including implementation level considerations such as data capturing from resolver/motor phase current ADCs, PWM switching, the motor and controller models. Following simulations, implementation and testing on the Infineon DSP HEMAS platform is to take place. In order to compare simulation and hardware test results the same system function is planned to be performed on each simulation and experimental step as per below two scenarios.

Simulation/Test scenario 1: this scenario aims to prove the key angle tracking equation 6.15. The scenario involves the motor spinning at low speed and a pulsating HFI to be applied based on a constant estimated angle $\hat{\vartheta}$. This scenario is repeated for 6 test cases each having a fixed $\hat{\vartheta}$, case1 $\hat{\vartheta}$=0, case2 $\hat{\vartheta}$= π/3, case3 $\hat{\vartheta}$=2π/3, case4 $\hat{\vartheta}$=π, case5 $\hat{\vartheta}$=4π/3, case6 $\hat{\vartheta}$=5π/3. Scenario1 is to be applied in all simulation and test steps and will illustrate how the saliency angle error varies as a function of $\Delta\vartheta$.

Simulation/Test scenario 2: this scenario intends to harvest the yields from scenario1. The angle tracking error analysed in the former scenario is used here to track the rotor angle of a spinning motor. This scenario is proposed to be applied in the VHDL simulation and hardware testing steps as the Matlab model is too abstract and does not contain detailed motor and controller models.

The following sections of this chapter, illustrate results at decreasing level of abstraction, starting from high level Matlab modelling, to low level VHDL simulation and final hardware testing on the HEMAS platform.

6.6 Matlab simulation

6.6.1 Matlab Scenario 1

Two Matlab models were developed aiming to simulate the saliency tracking method of scenario 1. The first model makes the idealistic assumption that the selected HEMAS selected saliency ($L_d = 1.069$mH $L_q = 1.158$mH) of section 6.4 is constant irrespective of the estimated angle $\hat{\vartheta}$. In practice however when $\hat{\vartheta}$ deviates from ϑ, less current will be deposited to the motor's d axis which in turn will reduce the stator iron saturation and decrease the motor's saliency. A second model was thus developed with the more realistic condition that the saliency of the motor is a function of i_d as per saliency in Table 6.1.

The Matlab source code for both the constant and varying saliency can be found in Appendix sections B.1.5 and B.1.6 of this thesis. The results of the two Matlab simulations are displayed one below the other for ease of comparison.

Matlab saliency simulation $\hat{\vartheta}$= 0 rad

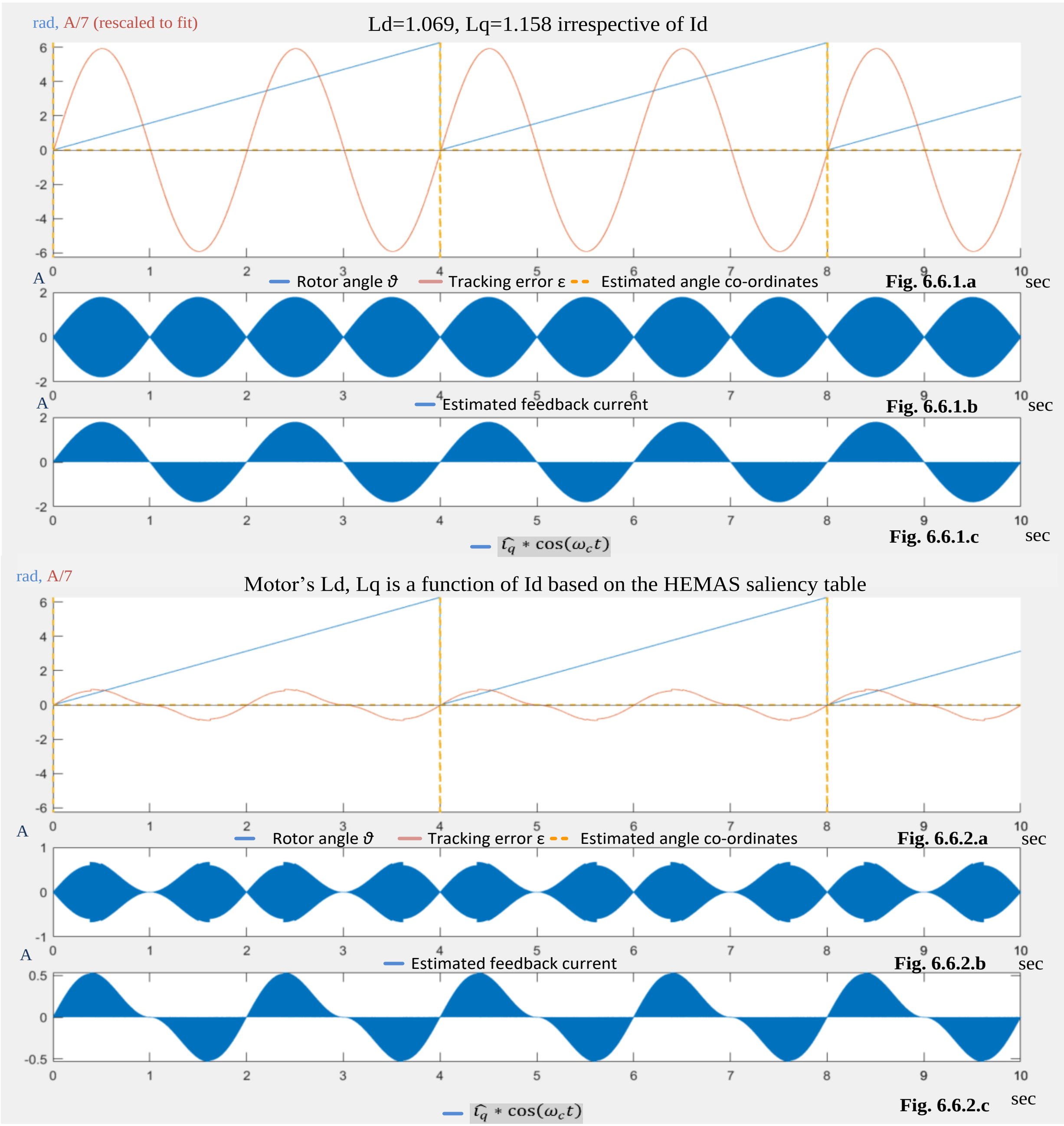


Figure 6.6 Matlab simulation of the saliency algorithm when rotor angle = 0 rad

Analysis of the Matlab simulation:

- The blue line of figures 6.6.1.a and 6.6.2.a illustrate the rotor electrical ϑ and therefore shows that the motor is spinning. The dotted orange line on the above-mentioned figures illustrate that the estimated angle $\hat{\vartheta}$ is constant with $\hat{\vartheta}$=0.

- The red line in figures 6.6.1.a.and 6.6.2.a is the saliency angle error as depicted in equation 6.15. This angle error is indeed a sinusoidal function of $2\Delta\vartheta$. Within one electrical period, there are two sinusoidal iterations of this angle error.
- The saliency error in Figure 6.6.2.a is about 3.5 times smaller in amplitude than in Figure 6.6.1.a. So the more representative simulation where i_d affects saliency shows that there is a rather large reduction in the angle tracking accuracy. This is because when $\Delta\vartheta = 0$ and saliency is maximum the saliency error is 0. As $\Delta\vartheta$ and saliency angle error increase the saliency decreases, so when the error is at its relative highest point, the saliency is at its lowest.
- Figures 6.6.1.b and 6.6.2.b illustrate the estimated q axis current $\widehat{\iota_q}$.
- Figures 6.6.1.c and 6.6.2.c illustrate the calculation of $\widehat{\iota_q} * \cos(\omega_c t)$. The application of a LPF to this calculation results into the saliency angle error depicted in figures 6.6.1.a and 6.6.2.a.
- what is of particular interest is that in the more representative simulation of L_d, L_q being a function of i_d (see Figure 6.7 bottom part showing a zoomed in version of Figure 6.6.1.b), there is a harmonic component in the $\widehat{\iota_q}$ relative to the simplified simulation where L_d, L_q are considered constant. This is because as i_d increases so does the saliency creating this additional harmonic in the estimated currents.

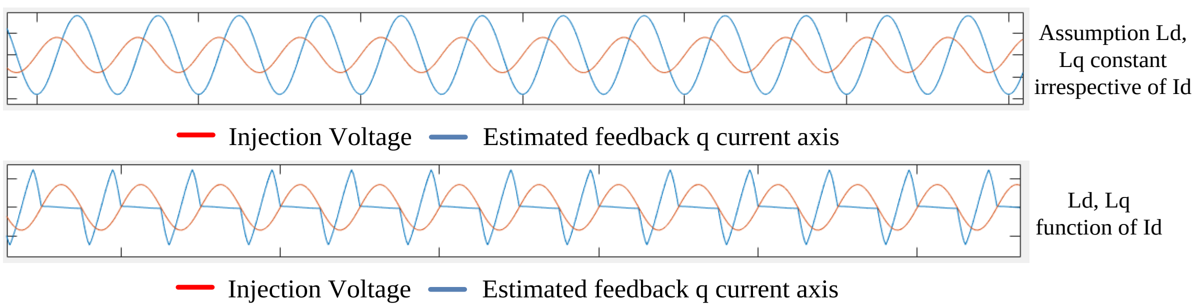


Figure 6.7 Matlab simulation of the saliency algorithm when rotor angle = 0 rad

Also note that as shown in Appendix section B.1.6, the Matlab model simulating varying saliency uses the existing saliency points from Table 6.1 and estimates the motor's inductance for any value of i_d between these points using linear interpolation as per pseudo code snapshot below:

```
Ld(x) = Ld(x-1) + (Ld(x+1) - Ld(x-1) * (Id(x) - Id(x-1))/(Id(x+1) - Id(x-1));

Lq(x) = Lq(x-1) + (Lq(x+1) - Lq(x-1) * (Id(x) - Id(x-1))/(Id(x+1) - Id(x-1));
```

Below are Matlab simulations for the additional cases of $\hat{\vartheta} = \pi/3$, $\hat{\vartheta} = 2\pi/3$, $\hat{\vartheta} = \pi$, $\hat{\vartheta} = 4\pi/3$, $\hat{\vartheta} = 5\pi/3$.

Matlab saliency simulation $\hat{\vartheta}$= $\pi/3$ rad

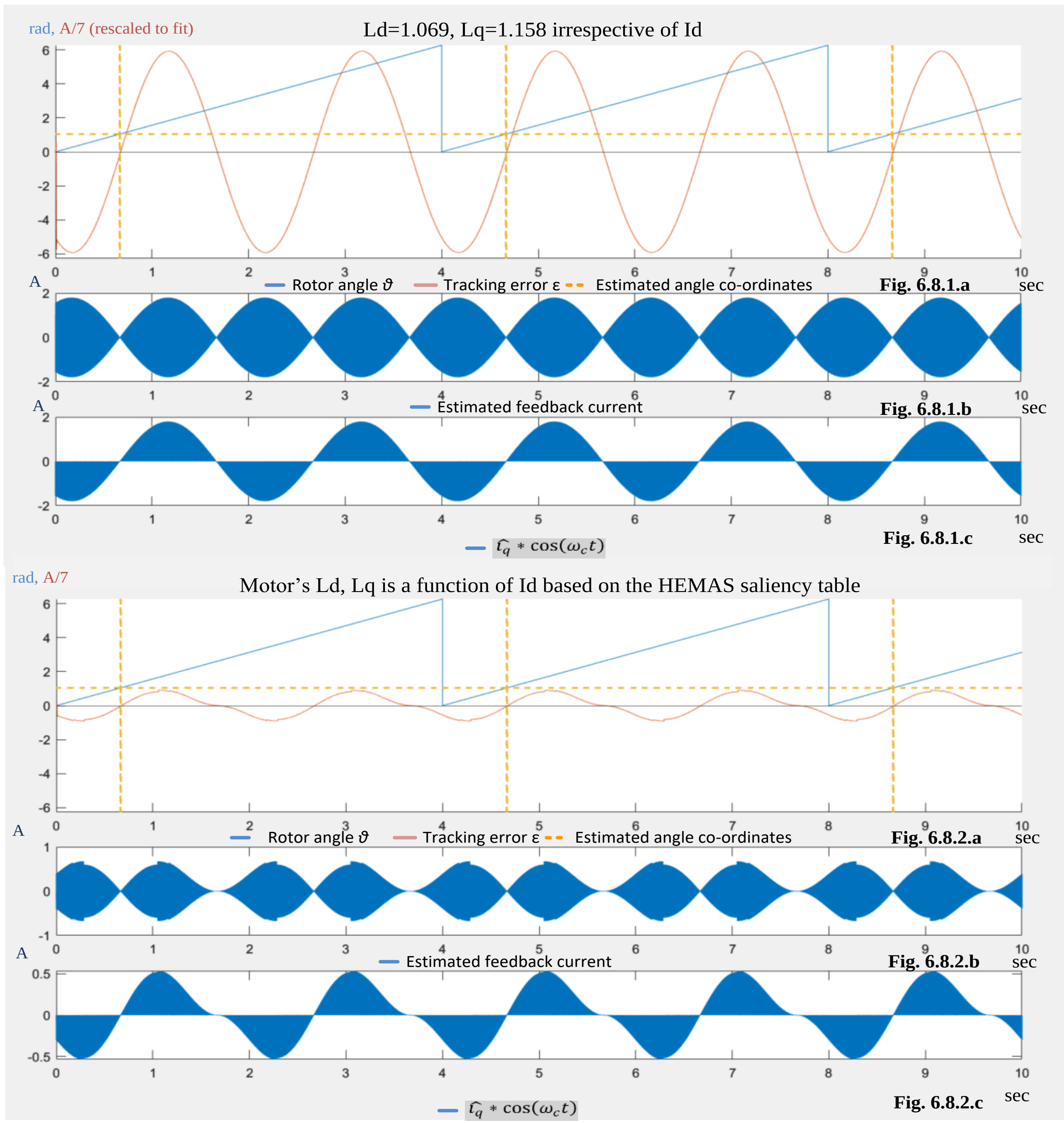


Figure 6.8 Matlab simulation of the saliency algorithm when rotor angle = $\pi/3$ rad

Analysis of Matlab simulation:

- Simulation results and related analysis are identical to that when $\hat{\vartheta}$=0, with the main difference being that the HFI and demodulation is based on assumption $\hat{\vartheta} = \pi/3$ rad. The sinusoidal angle tracking error in figure 6.8.1.a/6.8.2.a is therefore 0 when $\vartheta = \hat{\vartheta} = \pi/3$ and when $\vartheta = \hat{\vartheta} + \pi$.

Matlab saliency simulation $\hat{\vartheta}$= $2\pi/3$ rad

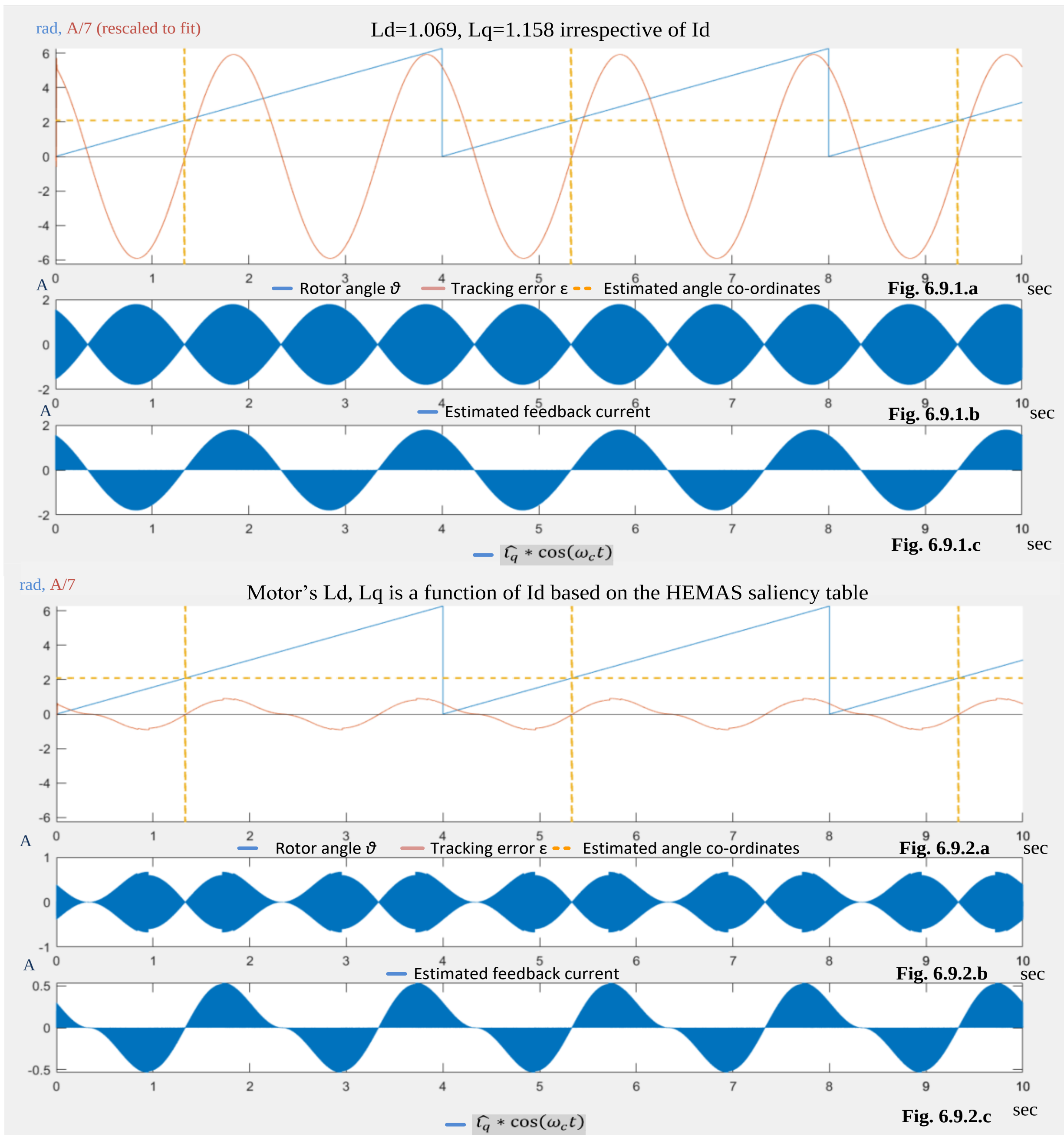


Figure 6.9 Matlab simulation of the saliency algorithm when rotor angle = $2 * \pi/3$ rad

Analysis of Matlab simulation:

- Simulation results and related analysis are identical to that when $\hat{\vartheta}$=0, with the main difference being that the HFI and demodulation is based on assumption $\hat{\vartheta} = 2\pi/3$ rad. The sinusoidal angle tracking error in figure 6.9.1.a/6.9.2.a is therefore 0 when $\vartheta = \hat{\vartheta} = 2\pi/3$ and when $\vartheta = \hat{\vartheta} + \pi$.

Matlab saliency simulation $\hat{\vartheta}$= $3\pi/3$ rad

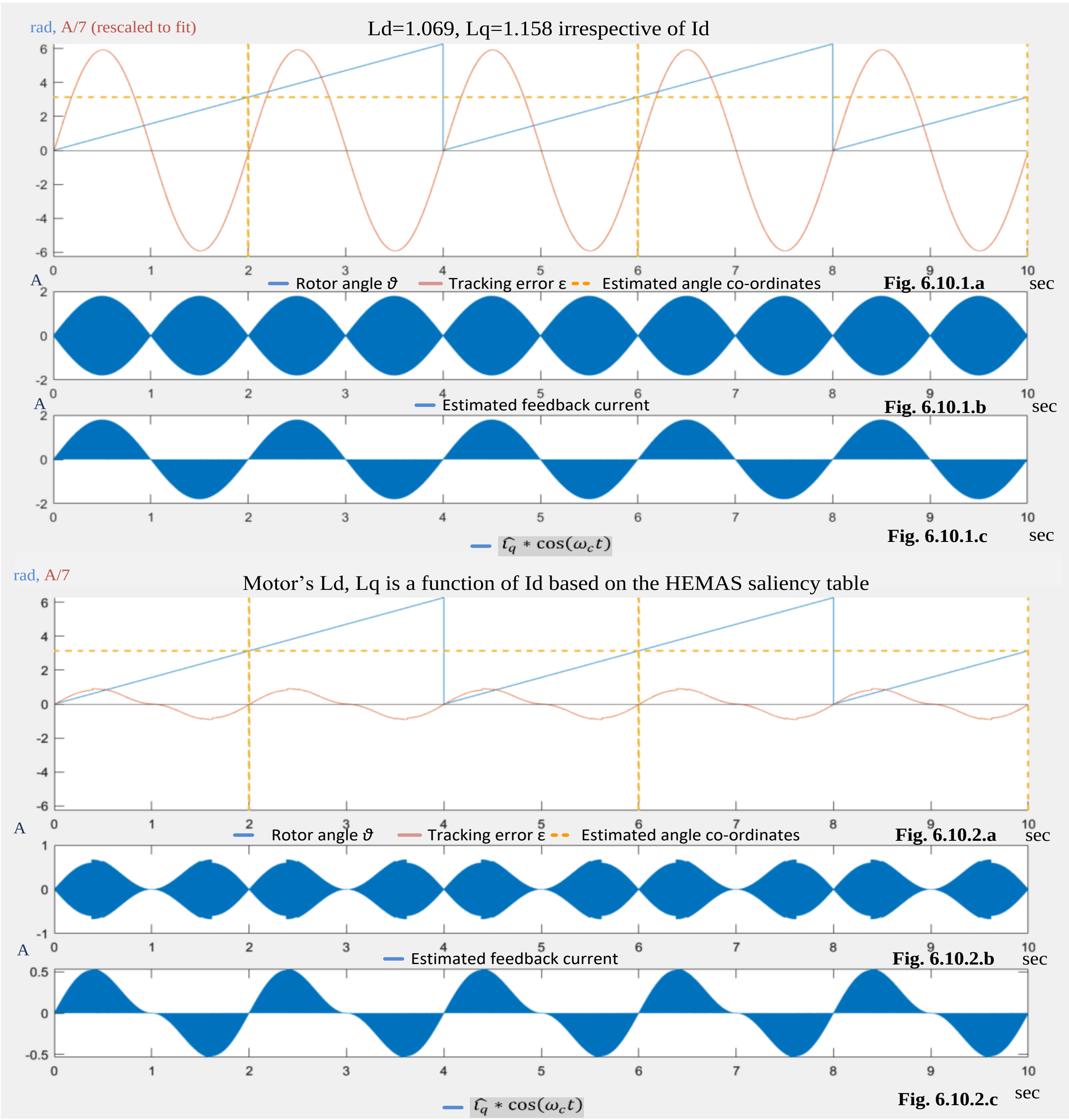


Figure 6.10 Matlab simulation of the saliency algorithm when rotor angle = $3 * \pi/3$ rad

Analysis of Matlab simulation:

- Simulation results and related analysis are identical to that when $\hat{\vartheta}$=0, with the main difference being that the HFI and demodulation is based on assumption $\hat{\vartheta} = 3\pi/3$ rad. The sinusoidal angle tracking error in figure 6.10.1.a/6.10.2.a is therefore 0 when $\vartheta = \hat{\vartheta} = 3\pi/3$ and when $\vartheta = \hat{\vartheta} + \pi$.

Matlab saliency simulation $\hat{\vartheta}$= $4\pi/3$ rad

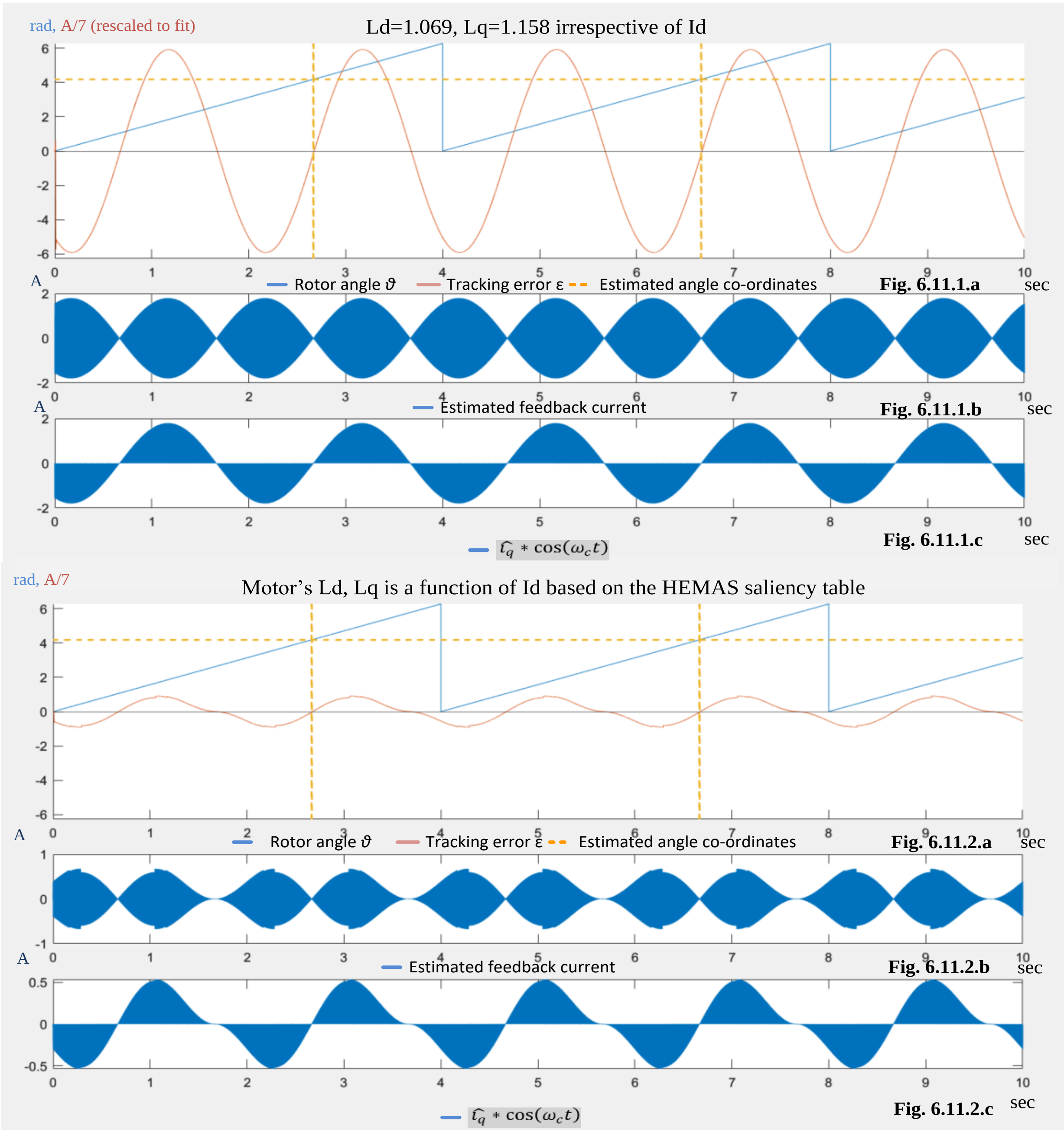


Figure 6.11 Matlab simulation of the saliency algorithm when rotor angle = $4 * \pi/3$ rad

Analysis of Matlab simulation:

- Simulation results and related analysis are identical to that when ϑ=0, with the main difference being that the HFI and demodulation is based on assumption $\hat{\vartheta} = 4\pi/3$ rad. The sinusoidal angle tracking error in figure 6.11.1.a/6.11.2.a is therefore 0 when $\vartheta = \hat{\vartheta} = 4\pi/3$ and when $\vartheta = \hat{\vartheta} + \pi$.

Matlab saliency simulation $\hat{\vartheta}$= 5π/3 rad

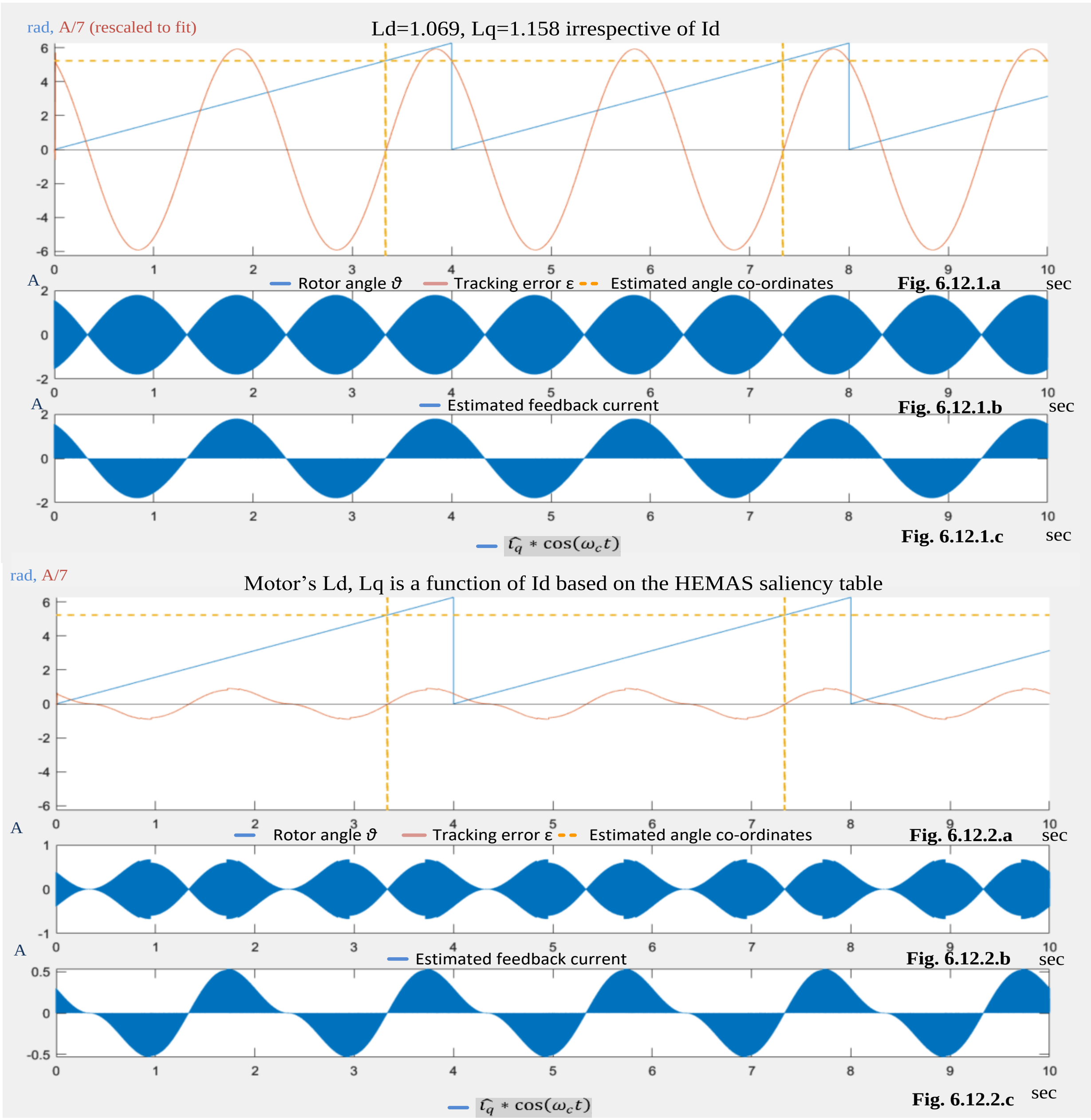


Figure 6.12 Matlab simulation of the saliency algorithm when rotor angle = 5 $*\pi/3$ rad

Analysis of Matlab simulation:

- Simulation results and related analysis are identical to that when $\hat{\vartheta}$=0, with the main difference being that the HFI and demodulation is based on assumption $\hat{\vartheta} = 5\pi/3$ rad. The sinusoidal angle tracking error in figure 6.12.1.a/6.12.2.a is therefore 0 when $\vartheta = \hat{\vartheta} = 5\pi/3$ and when $\vartheta = \hat{\vartheta} + \pi$.

6.7 VHDL simulation

The HFI demodulation algorithm was simulated in a VHDL simulation environment at low level of implementation. The purpose of this simulation as stated in section 6.5 is to evaluate the algorithm at low level of implementation taking into account hardware considerations. The simulation includes capturing data from ADC sensors, a resolver model, processing data within the controller including speed and current control loops, forward/inverse Clarke/Park transforms, Space Vector Modulation (SVM), PWM switching, an inverter and motor model. All of these RTL functional blocks along with the HFI sensorless block as shown in Figure 6.13 are active during this simulation. Note that the motor model parameters align with the HEMAS parameters. As per simulation plan, two simulation scenarios are presented one that illustrates the saliency angle error (scenario 1) and one where the HFI sensorless demodulation method is used to track the rotor angle while spinning (scenario 2). In scenario1, both simulation results of considering L_d, L_q constant or a function of i_d are presented similarly to Matlab.

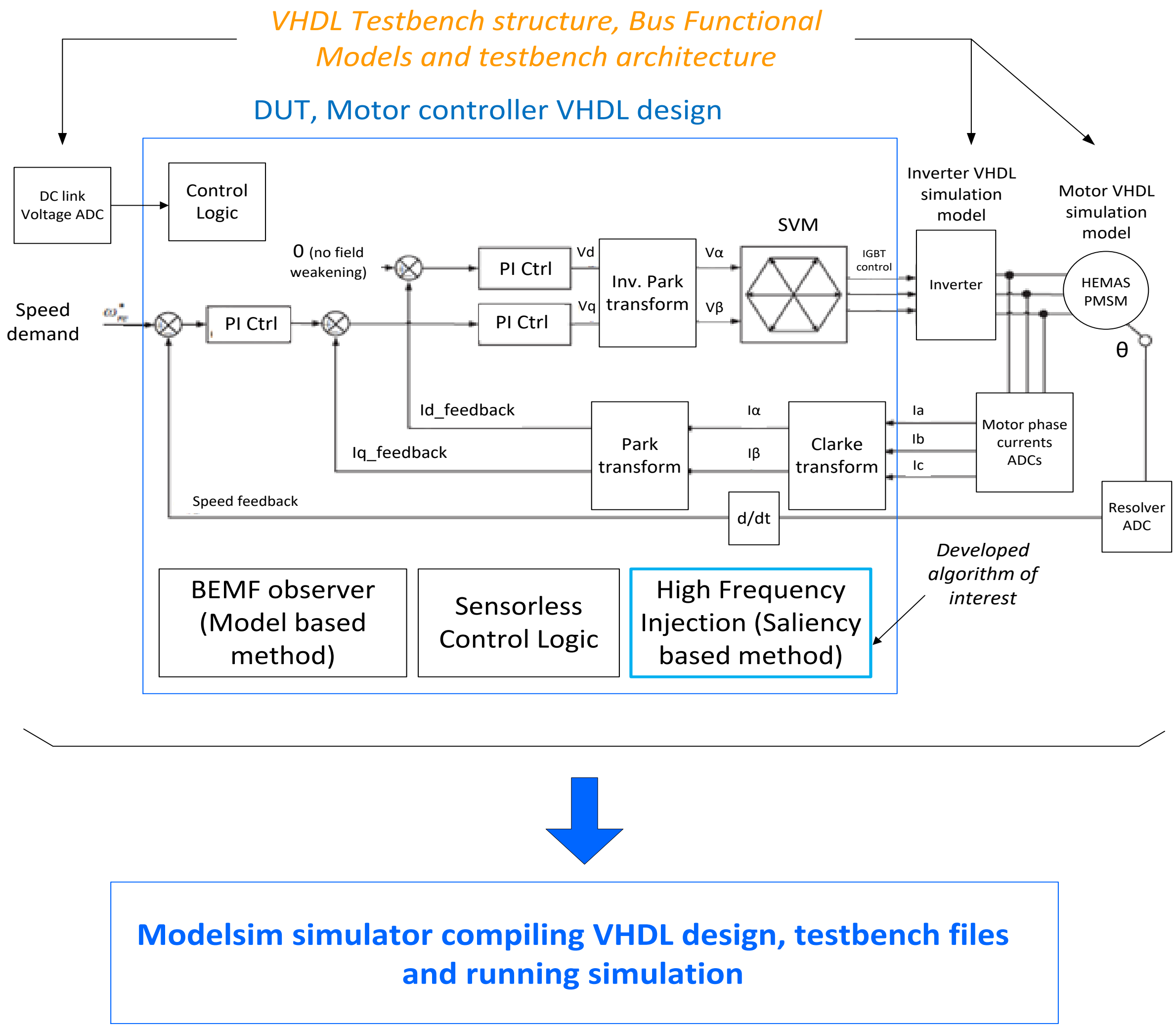
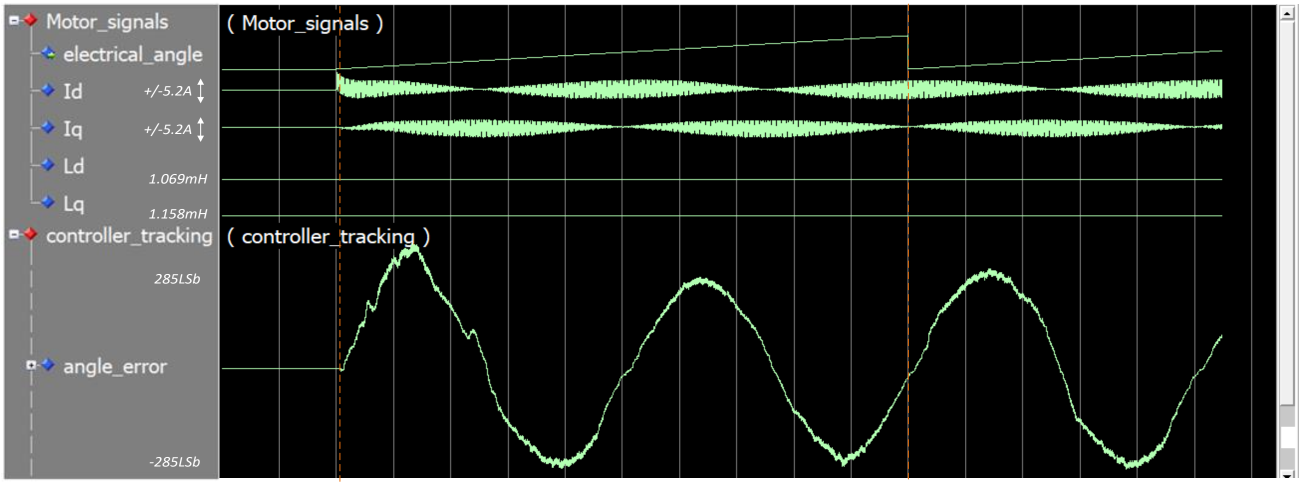


Figure 6.13 VHDL simulation structure

6.7.1 VHDL Scenario1

VHDL saliency simulation $\hat{\vartheta}$= 0 rad

Ld=1.069, Lq=1.158 irrespective of Id



Motor's Ld, Lq is a function of Id based on the HEMAS saliency table

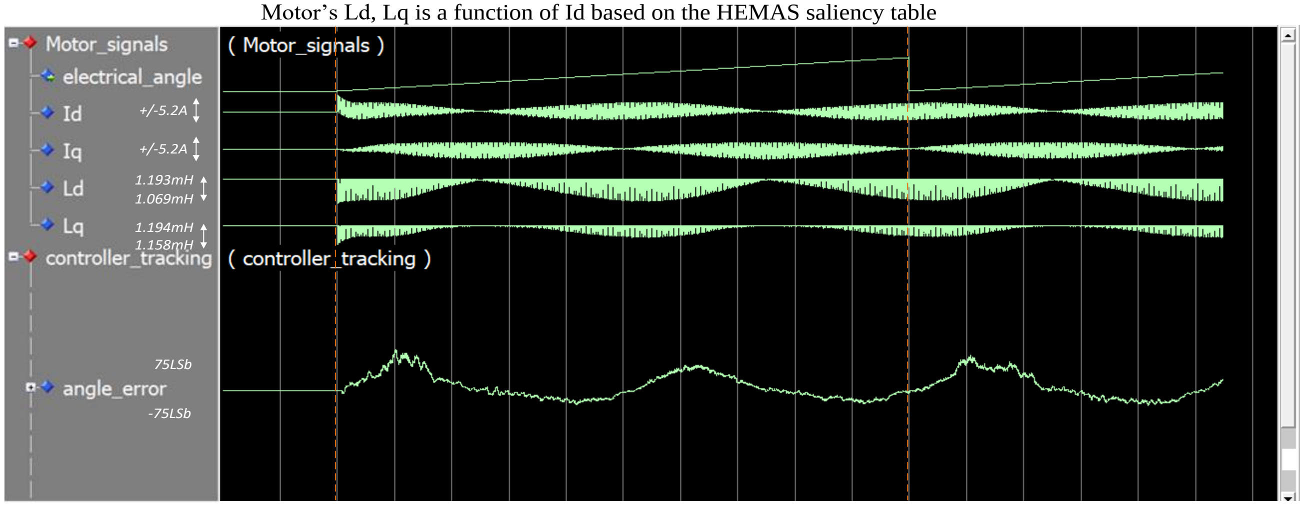
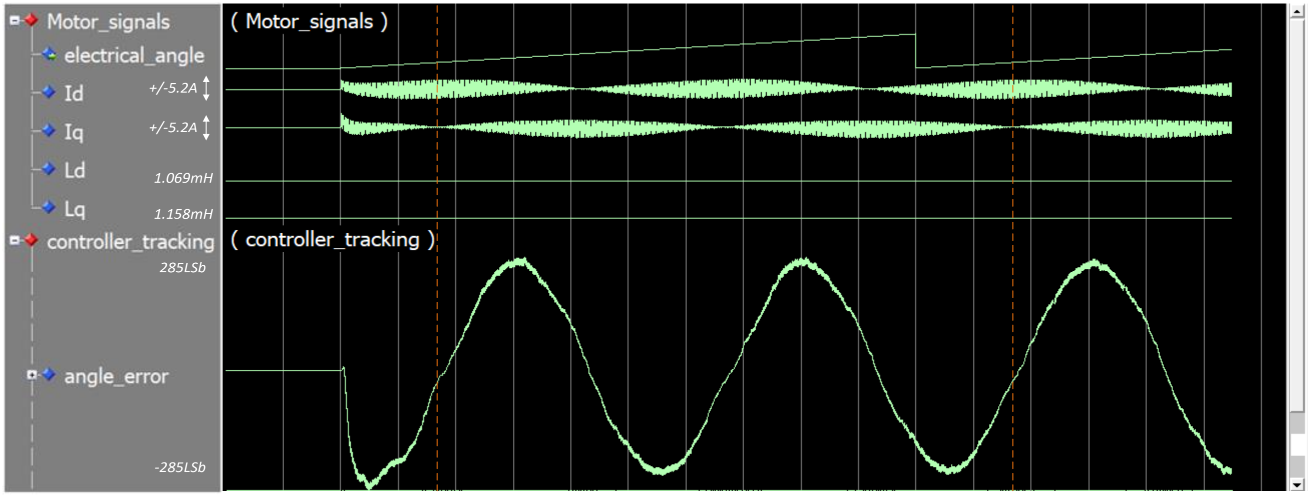


Figure 6.14 VHDL simulation of the saliency algorithm when $\hat{\vartheta}$= 0 rad and the motor is spinning

VHDL saliency simulation $\widehat{\vartheta} = \pi/3$ rad

Ld=1.069, Lq=1.158 irrespective of Id



Motor's Ld, Lq is a function of Id based on the HEMAS saliency table

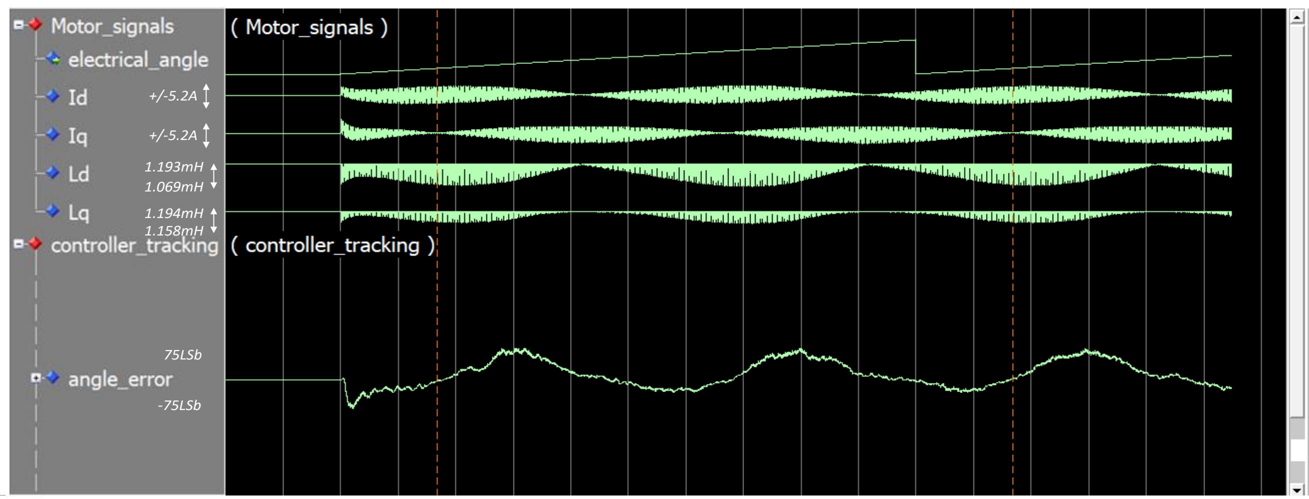
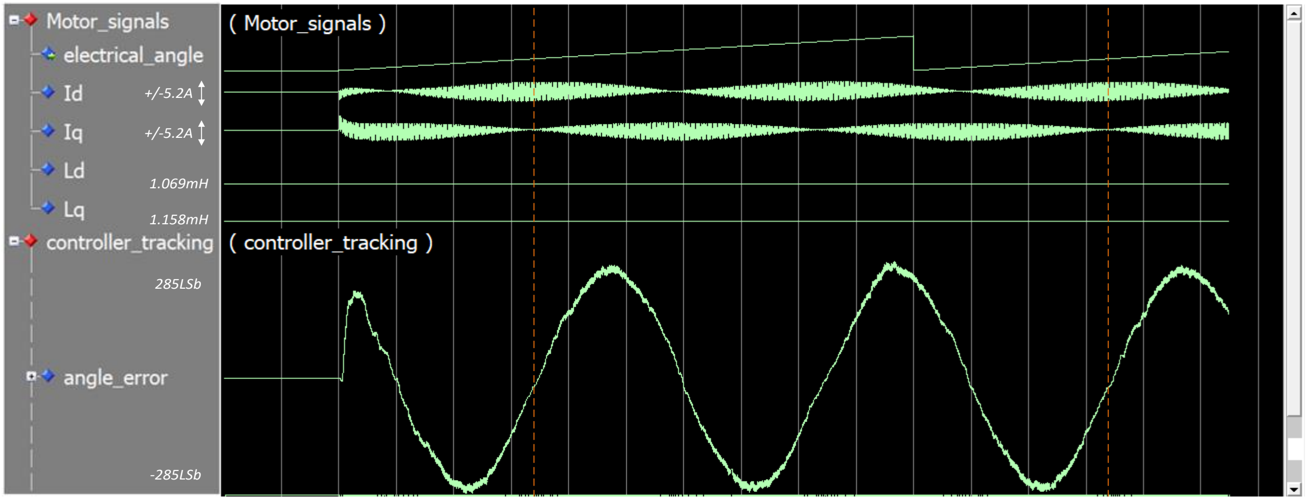


Figure 6.15 VHDL simulation of the saliency algorithm when $\hat{\vartheta} = \pi/3$ rad and the motor is spinning

VHDL saliency simulation $\hat{\vartheta}$= 2π/3 rad

Ld=1.069, Lq=1.158 irrespective of Id



Motor's Ld, Lq is a function of Id based on the HEMAS saliency table

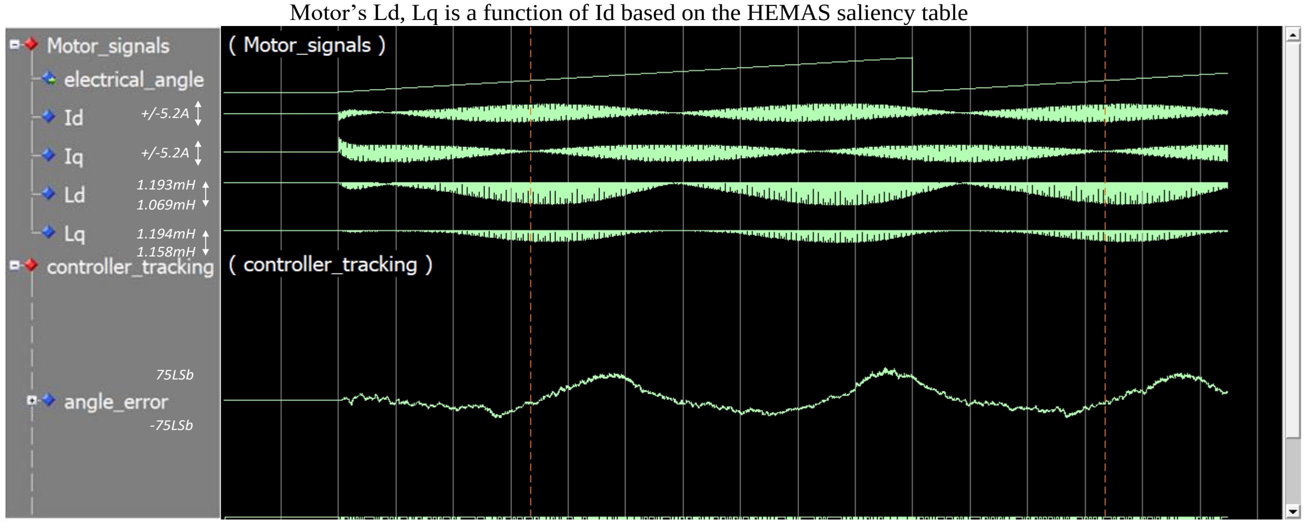
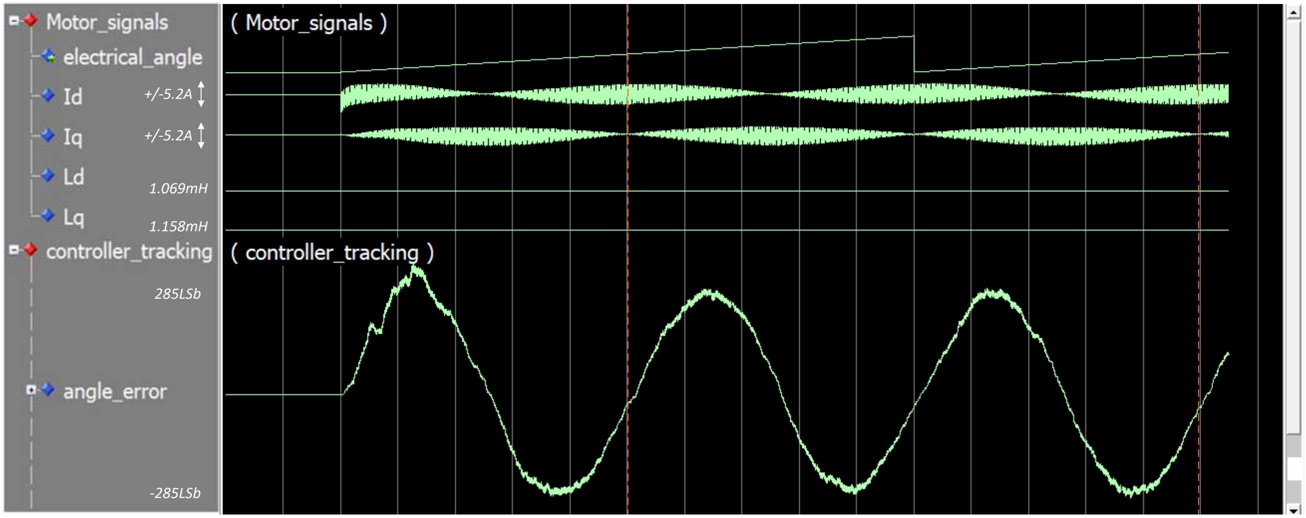


Figure 6.16 VHDL simulation of the saliency algorithm when $\hat{\vartheta} = 2\pi/3$ rad and the motor is spinning

VHDL saliency simulation $\hat{\vartheta}$= $3\pi/3$ rad

Ld=1.069, Lq=1.158 irrespective of Id



Motor's Ld, Lq is a function of Id based on the HEMAS saliency table

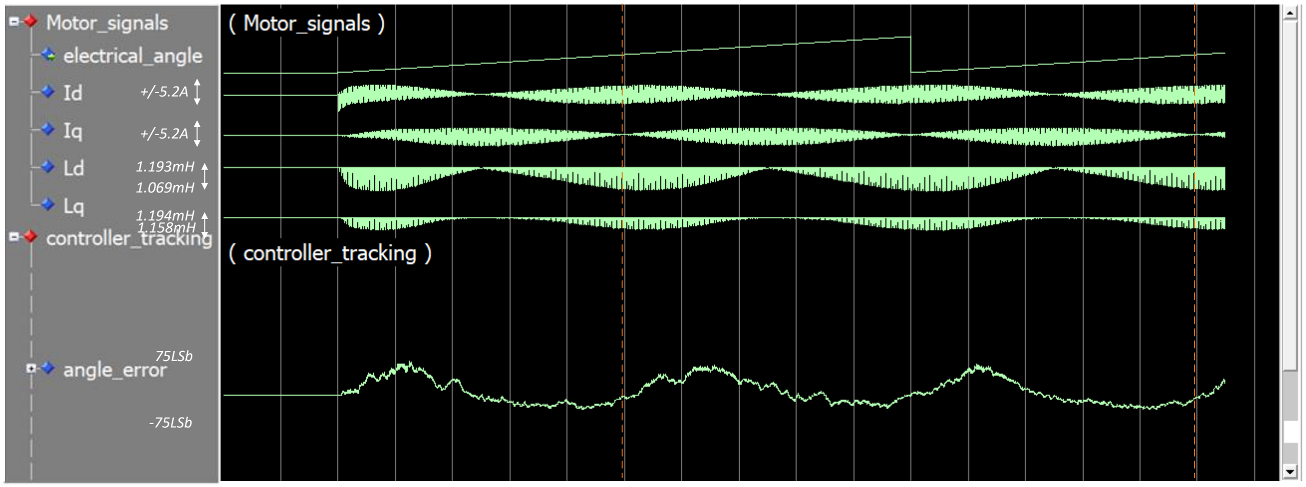
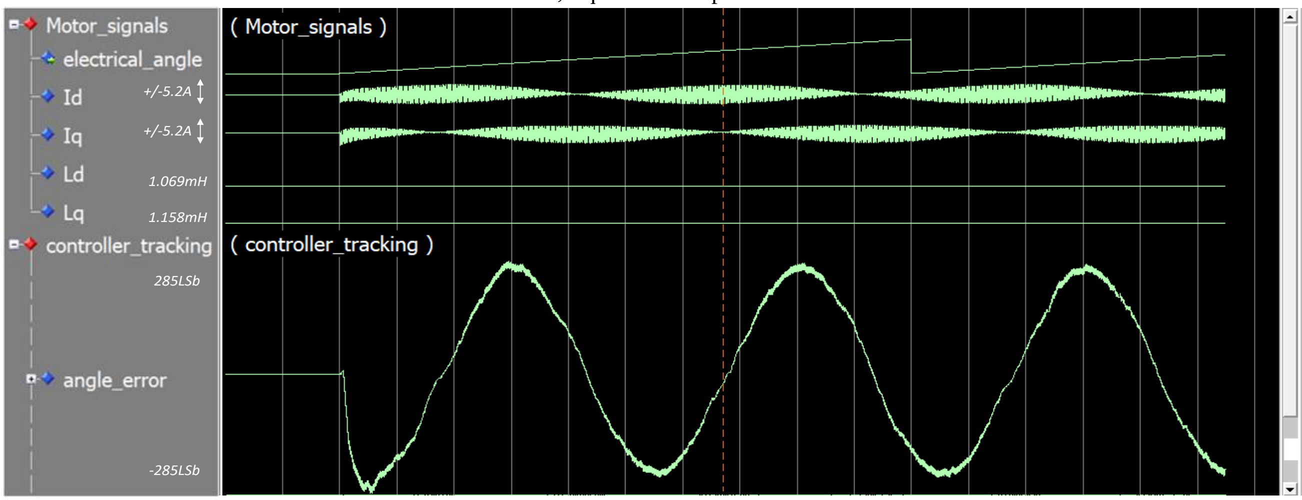


Figure 6.17 VHDL simulation of the saliency algorithm when $\hat{\vartheta} = 3\pi/3$ rad and the motor is spinning

VHDL saliency simulation $\hat{\vartheta}$= $4\pi/3$ rad

Ld=1.069, Lq=1.158 irrespective of Id



Motor's Ld, Lq is a function of Id based on the HEMAS saliency table

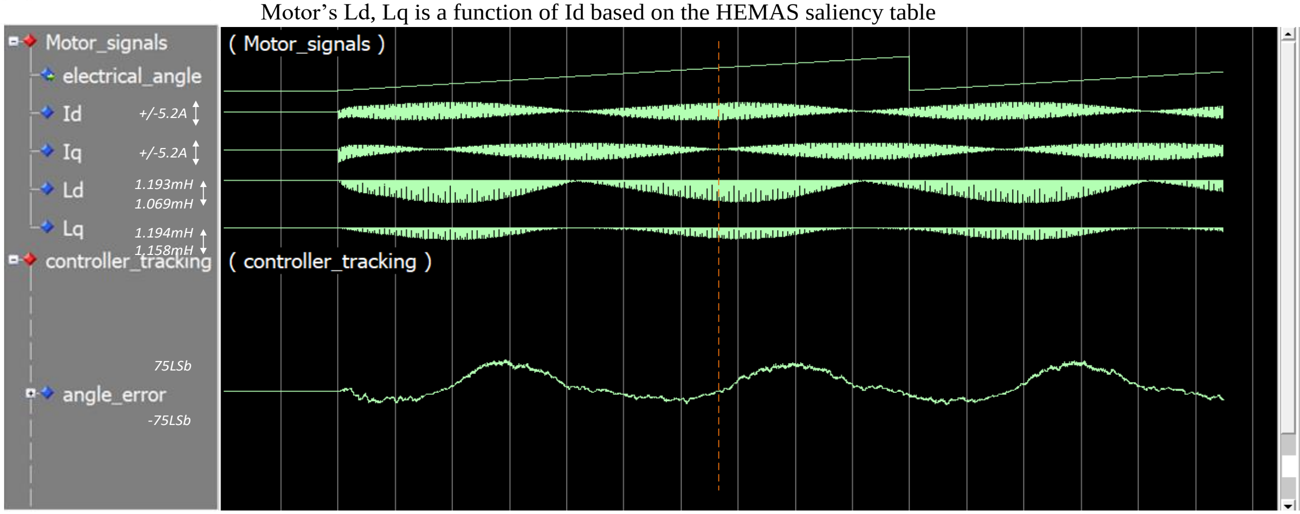
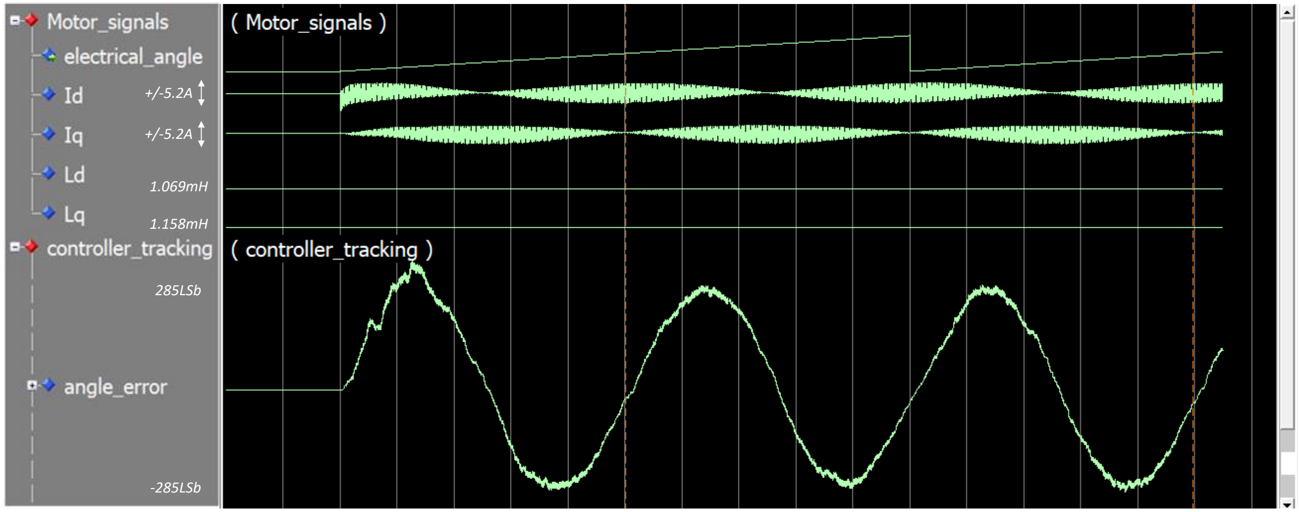


Figure 6.18 VHDL simulation of the saliency algorithm when $\hat{\vartheta} = 4\pi/3$ rad and the motor is spinning

VHDL saliency simulation $\hat{\vartheta}$= $5\pi/3$ rad

Ld=1.069, Lq=1.158 irrespective of Id



Motor’s Ld, Lq is a function of Id based on the HEMAS saliency table

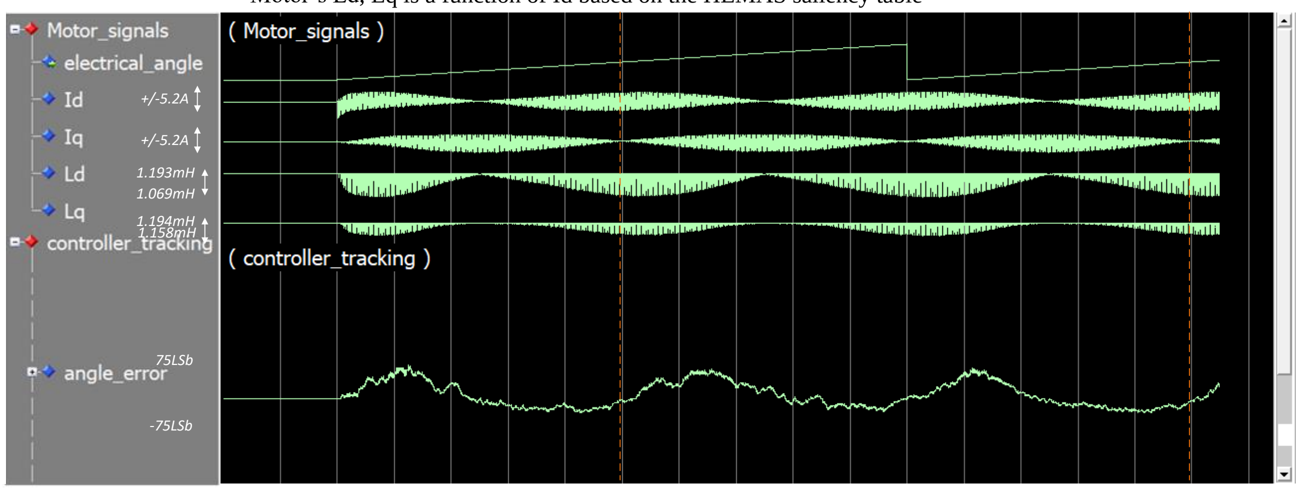


Figure 6.19 VHDL simulation of the saliency algorithm when $\hat{\vartheta} = 5\pi/3$ rad and the motor is spinning

Analysis of VHDL simulation for $\hat{\vartheta}$= 0 rad:

- The angle tracking error signal "angle_error" is as expected of substantially lower amplitude when Lq/Ld varies as a function of i_d as shown in figure 6.14.bottom versus 6.14.top. The angle tracking error signal as expected is 0 when $\vartheta = \hat{\vartheta} = 0$ and when $\vartheta = \hat{\vartheta} + \pi$.
- "angle_error" has a transient when injection starts, this is because of HPF, LPF take some time to stabilize.
- "angle_error" in both idealistic and realistic assumptions (6.14 top/bottom) is more spiky, noisy and of less of sinusoidal shape versus equivalent Matlab simulations (Figure 6.6). This is because low level implementation such as PWM and ADC quantization have an impact on the overall accuracy. Note that the sinusoid that is multiplied to the feedback current $\widehat{\iota_q}$ includes a configurable offset to accommodate for propagation delay of the HF voltage reaching the motor and received back as HFI current by the controller. Matlab simulation does not model this propagation delay so there is not a need to have this offset compensation.

Overall the VHDL simulation results align with Matlab simulations in terms of tracking error signal but VHDL simulations suffer with implementation level side effects such as propagation delay of signals to the motor and back to the controller via ADCs, LPF/HPF propagation delay, and quantization effects.

The analysis for the subsequent VHDL simulation results shown in Figures 6.15-6.19 for estimated angle $\hat{\vartheta}$= $\pi/3$ rad, $\hat{\vartheta}$= $2\pi/3$ rad, $\hat{\vartheta}$= $3\pi/3$ rad, $\hat{\vartheta}$= $4\pi/3$ rad, $\hat{\vartheta}$= $5\pi/3$ rad, is identical to the above with the main difference that the comments relate to the estimated angle of interest. For example regarding Figure 6.15 where $\hat{\vartheta}$= $\pi/3$ rad:

- The angle tracking error signal as expected is 0 when $\vartheta = \hat{\vartheta} = \pi/3$ rad and when $\vartheta = \hat{\vartheta} + \pi = \pi/3 + \pi$ rad. Similarly to Matlab simulation, the "angle_error" in the representative simulation of Figure 6.15 bottom is of lower amplitude than idealistic constant saliency simulation as Lq/Ld varies as a function of the current deposited to the d axis i_d.
- "angle_error" has a transient when injection starts, this is because of HPF, LPF take some time to stabilize.
- "angle_error" in both idealistic and realistic assumptions (6.15 top/bottom) is more spiky, noisy and of less of sinusoidal shape versus equivalent Matlab simulations (Figure 6.8).

6.7.2 VHDL Scenario2

Scenario 1 of VHDL simulations aimed to validate the angle tracking error signal. Having identified the shape and characteristics of this signal it is now possible to use this to track the rotor angle using a PI controller. This is done in scenario 2 of VHDL simulations shown below. The motor is therefore spinning and the saliency angle error is used to track the angle. One thing to note is that in VHDL simulation it is this saliency angle that is also used to drive the motor and not the resolver.

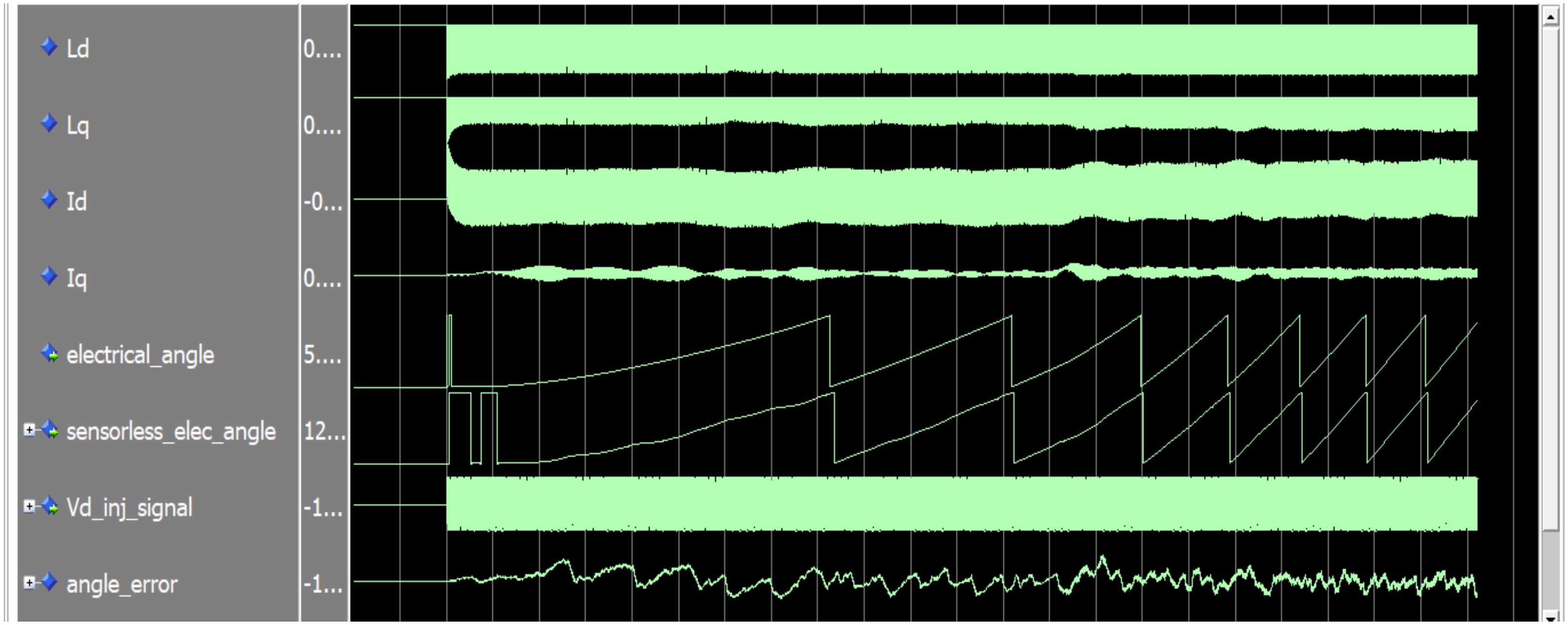


Figure 6.20 VHDL simulation, motor is spinning and HFI saliency error used to track rotor angle

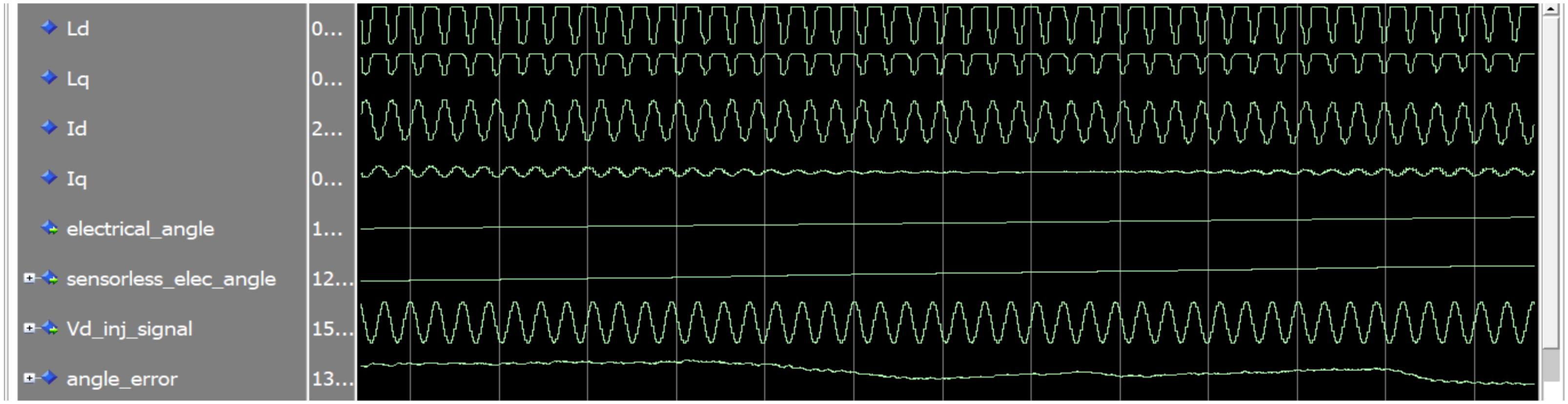


Figure 6.21 Previous figure zoomed in illustrating L_d, L_q variation as I_d changes due to angular error

Analysis of VHDL simulation scenario 2:

- L_q, L_d follow the realistic assumption that they are a function of i_d as per table 6.1 and using Linear interpolation as shown in Appendix B.3.1 of this thesis for any point in between the samples points of the table. In other words, as the controller's estimated angle aligns with the actual angle, all the HFI current is deposited to the actual d axis of the motor, resulting into optimal saliency points as per Table6.2. As the controller's estimated angle lags or advances versus the actual angle, the saliency is lower than its optimal point identified in Table 6.2.
- The angle tracking PI controller is continuously tracking the rotor angle. When the angle error increases, this results into some of the injected voltage to be deposited to the motor's q axis resulting into reduction of saliency (change of L_q, L_d) as shown in Figures 6.20 and 6.21.

6.8 Implementation of the algorithm on the HEMAS platform

The saliency method has been simulated in previous sections at different levels of abstraction. Having obtained confidence from these simulations the saliency observer code was implemented in C for the target Infineon DSP. The saliency method involves two key C functions, one to inject the HFI voltage to the estimated d axis (named PRHFI_gen) and a second to demodulate the HF feedback currents (named PRHFI_demod_angle). The C source code for the saliency based method can be found in section B.2.2 of this thesis.

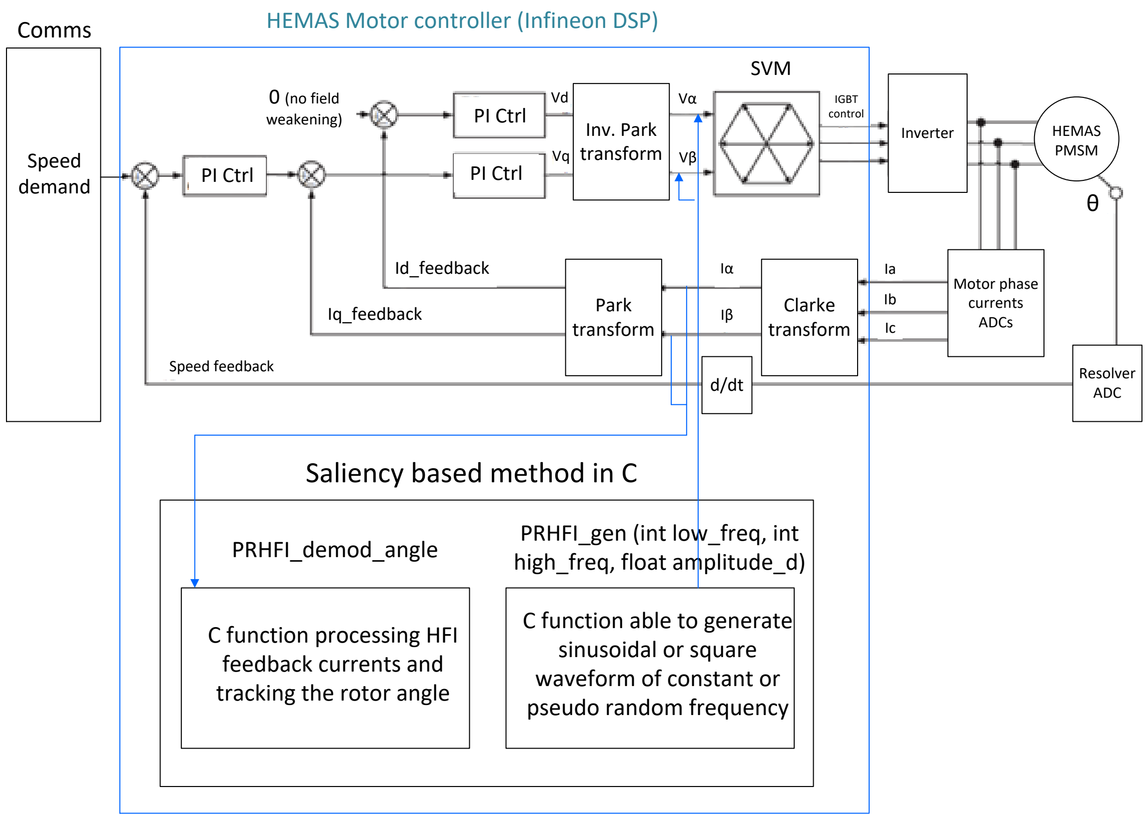


Figure 6.22 High Frequency Injection Block diagram

Note1: the C function "PRHFI_gen(int low_freq, int high_freq, float amplitude_d)" allows the user to modify the frequency and the shape (sinusoidal or square) of the injected voltage. While the function contains this flexibility for tests presented in the following chapters, in this chapter it is used to inject sinusoidal voltage of constant frequency and specifically at 1kHz by setting the lower and upper frequency thresholds at the same value as shown in Figure 6.23.

Note2: at first sight of Figure 6.22 the reader would observe that the HF injection is applied to the alpha/beta axis while it should have been applied to the d axis. This was done intentionally as in this test the resolver is still used to drive the motor based on resolver angle ϑ and the HFI has its own independent estimated angle $\hat{\vartheta}$. A straightforward way to maintain two independent domains ϑ and $\hat{\vartheta}$ where the former is used for driving the motor and the latter for the saliency observer is to translate the estimated d axis injection to the stationary axis as shown below in blue:

```
Valpha3_ref = Vd3_ref_dt*cosTheta3 - Vq3_ref_dt * sinTheta3 +
PRHFI_injection_d * cosThetasens3;

Vbeta3_ref = Vd3_ref_dt * sinTheta3 + Vq3_ref_dt * cosTheta3 +
PRHFI_injection_d * sinThetasens3;
```

where Theta3 = ϑ i.e. the resolver rotor angle and Thetasens3 = $\hat{\vartheta}$ is the saliency estimated rotor angle.

6.9 Hardware testing

6.9.1 Test Scenario 1

Similarly to the simulation, hardware testing scenario1 involves fixing the estimated angle to: case1 $\hat{\vartheta}$=0, case2 $\hat{\vartheta}$=π/3, case3 $\hat{\vartheta}$=2π/3, case4 $\hat{\vartheta}$=π, case5 $\hat{\vartheta}$=4π/3, case6 $\hat{\vartheta}$=5π/3 while the motor is spinning. The above estimated angle test points are repeated for two motor speeds: 15Hz (180 rpm) and 20Hz (240 rpm) where the controller is using the resolver interface to spin the motor. The HFI is set to 1kHz and of sinusoidal shape. The maximum injected current at the d axis is set to ≈ 5.2A to establish maximum saliency.

Data capturing of the internal DSP registers listed below is performed for analysis of test results:

- Resolver angle ϑ this is named in the HEMAS C code as Theta3.
- Motor phase currents I_a,I_b,I_c.
- The saliency angle error calculated in the Infineon DSP based on $(LPF(\widehat{i_q} * \cos(\omega_c t))$.
- High Frequency injection demodulation signal.

The above data was captured using the *.csv export facility of the Infineon DSP software saving 1000 samples of each register at a sampling rate of 10kHz i.e. once every 100 µs. The PWM switching frequency and the processing/control loops also function at 10kHz (once every 100 µs) so the sinusoidal HFI is composed of 10 sample points.

The *.csv data was not processed in Excel as in previous chapter but imported directly to Matlab in order to facilitate the more advanced processing and display capabilities of the tool. This was useful for example illustrating how the HF voltage is deposited to the actual d/q axis as the motor spins. The motor currents i_d and i_q are calculated within the Matlab processing script using the captured motor phase currents (I_a,I_b,I_c), the resolver angle ϑ and the Clarke/Park transform equations detailed in Appendix C of this thesis. The matlab scripts used to process the DSP data can be found in the Appendix section B.1.7 of this thesis.

Note that Figure 6.23 depicts the settings that the user can modify while the DSP code is running to modify the HFI injection frequency, shape, amplitude as well as the phase offset of the demodulation sinusoid that is used to multiply $\hat{i_q}$ to calculate the angular error.

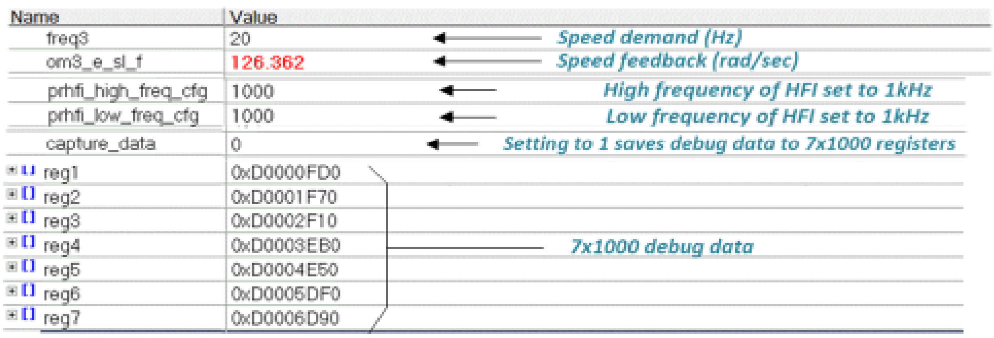


Figure 6.23 Infineon hardware test parameter set-up and results data

<u>Hardware testing scenario1 $\hat{\vartheta}$= 0 rad and motor spinning at 15Hz</u>

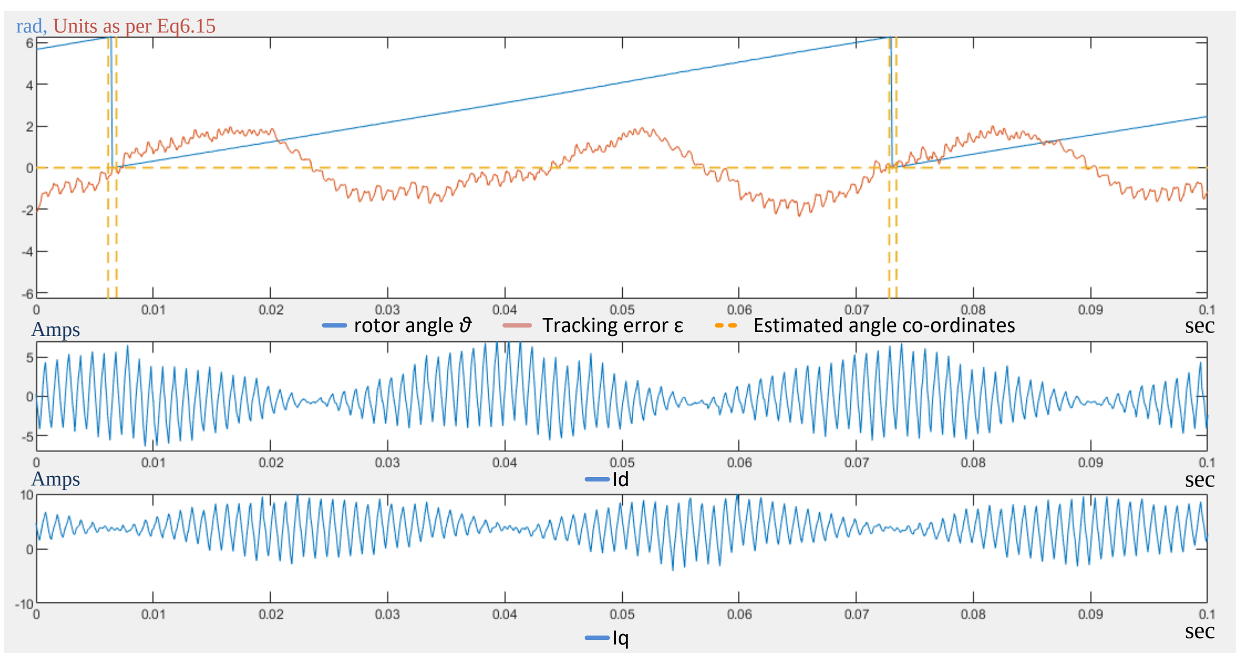


Figure 6.24 Hardware testing, saliency error when $\hat{\vartheta} = 0$ rad and the motor is spinning at 15Hz

Hardware testing scenario1 $\hat{\vartheta}$= $\pi/3$ rad and motor spinning at 15Hz

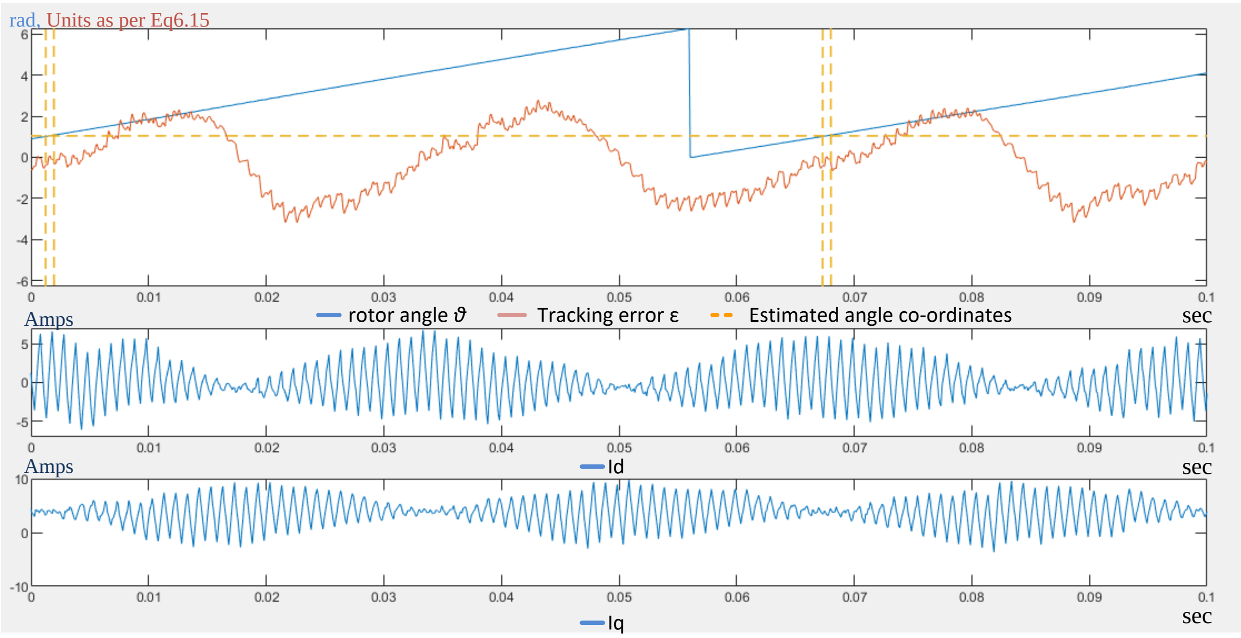


Figure 6.25 Hardware testing, saliency error when $\hat{\vartheta} = \pi/3$ rad and the motor is spinning at 15Hz

Note that Figures 6.24 and 6.25 illustrate how the algorithm tracks the rotor angle at two different captured windows. The test conditions are the same in both windows. The two illustrations aim to provide a wider sample window towards evaluating the algorithm.

Hardware testing scenario1 $\hat{\vartheta}$= $2\pi/3$ rad and motor spinning at 15Hz

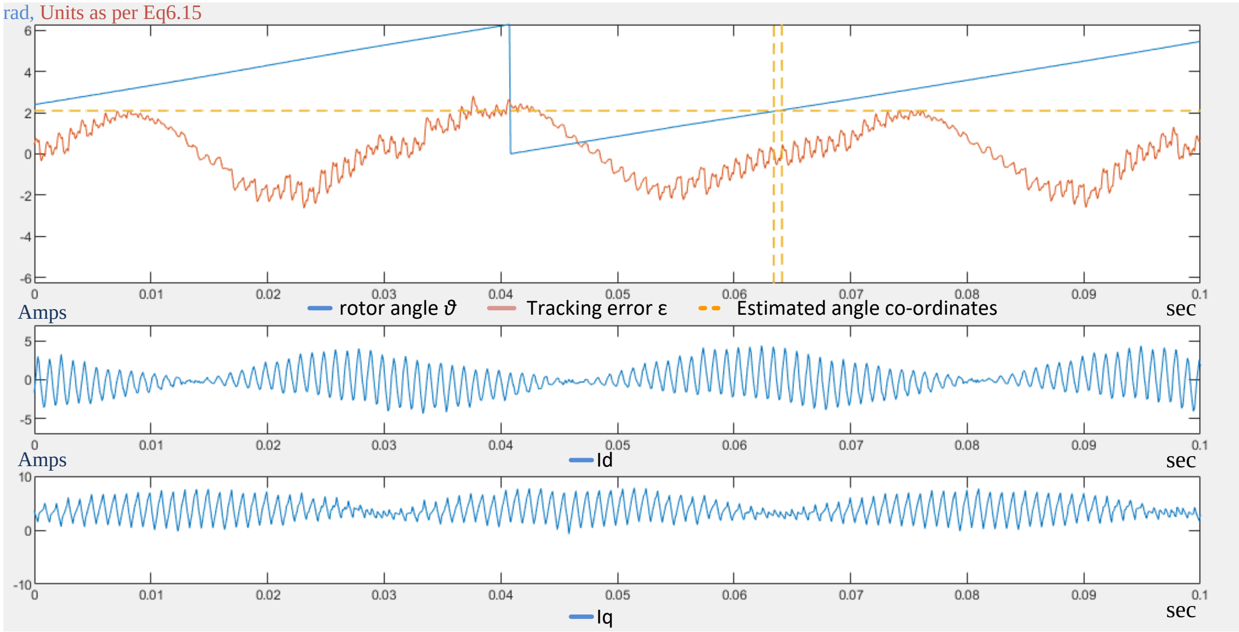


Figure 6.26 Hardware testing, saliency error when $\hat{\vartheta} = 2\pi/3$ rad and the motor is spinning at 15Hz

Hardware testing scenario1 $\hat{\vartheta}$= $3\pi/3$ rad and motor spinning at 15Hz

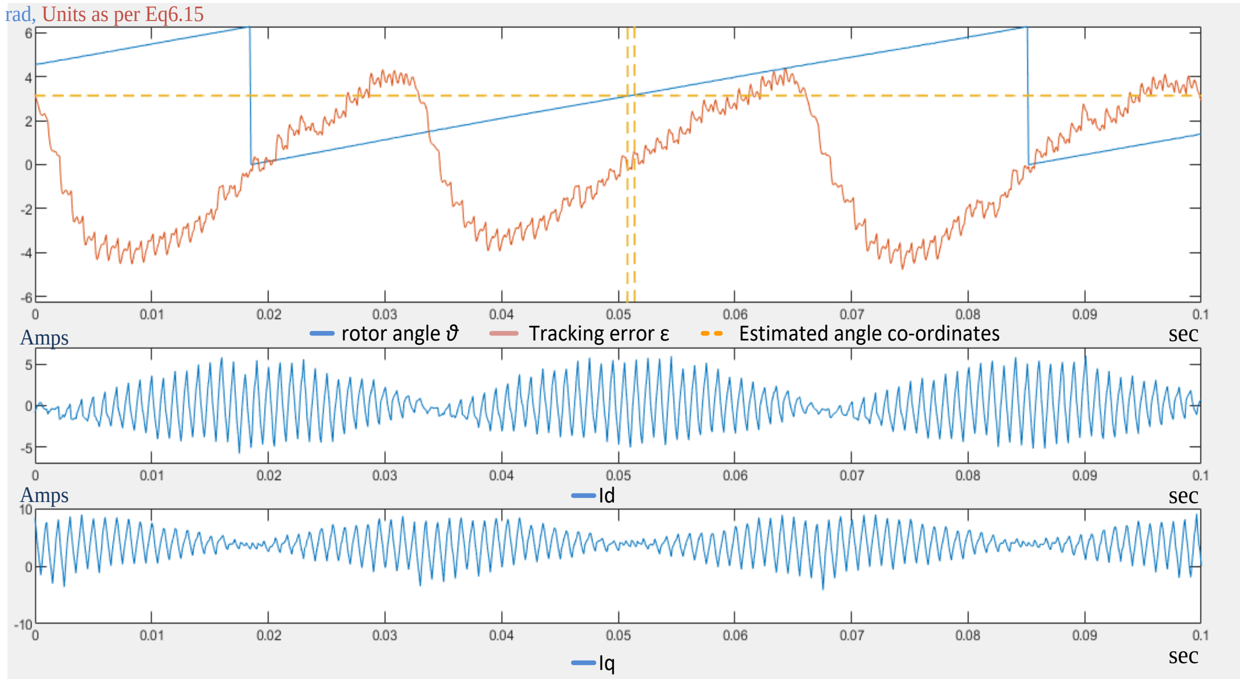


Figure 6.27 Hardware testing, saliency error when $\hat{\vartheta}$ $= 3\pi/3$ rad and the motor is spinning at 15Hz

Hardware testing scenario1 $\hat{\vartheta}$= $4\pi/3$ rad and motor spinning at 15Hz

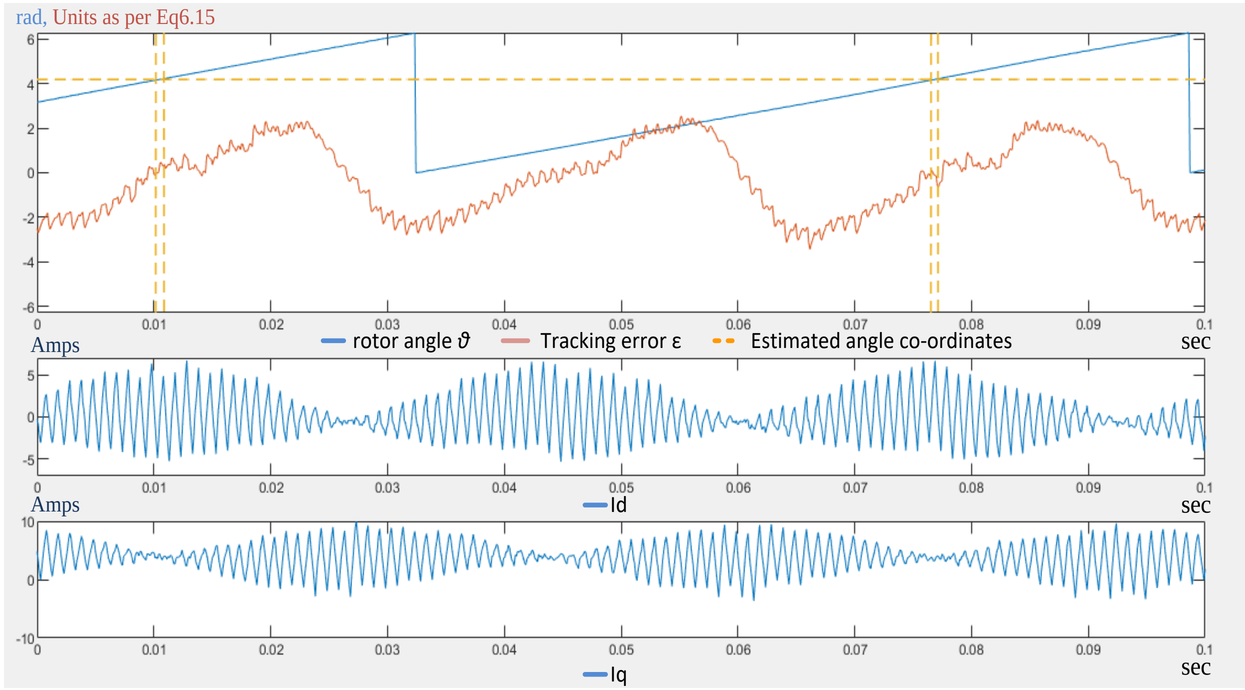


Figure 6.28 Hardware testing, saliency error when $\hat{\vartheta}$ $= 4\pi/3$ rad and the motor is spinning at 15Hz

Hardware testing scenario1 $\hat{\vartheta}$= 5π/3 rad and motor spinning at 15Hz

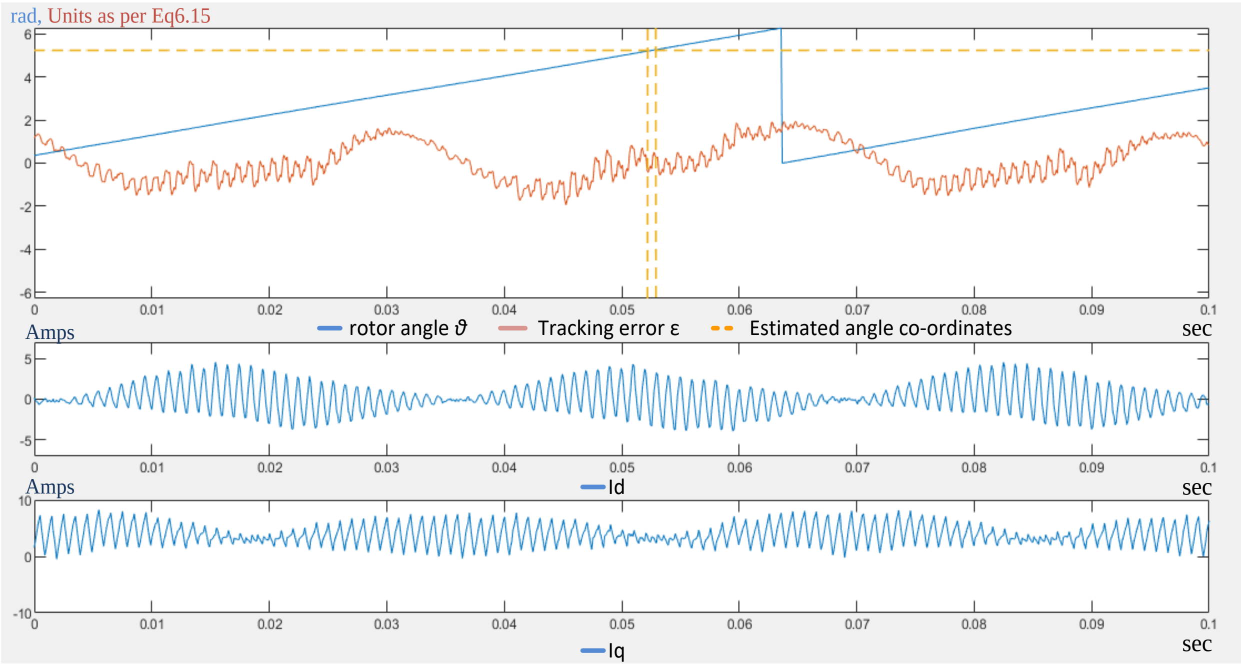


Figure 6.29 Hardware testing, saliency error when $\hat{\vartheta} = 5\pi/3$ rad and the motor is spinning at 15Hz

Hardware testing scenario1 $\hat{\vartheta}$= 0 rad and motor spinning at 20Hz

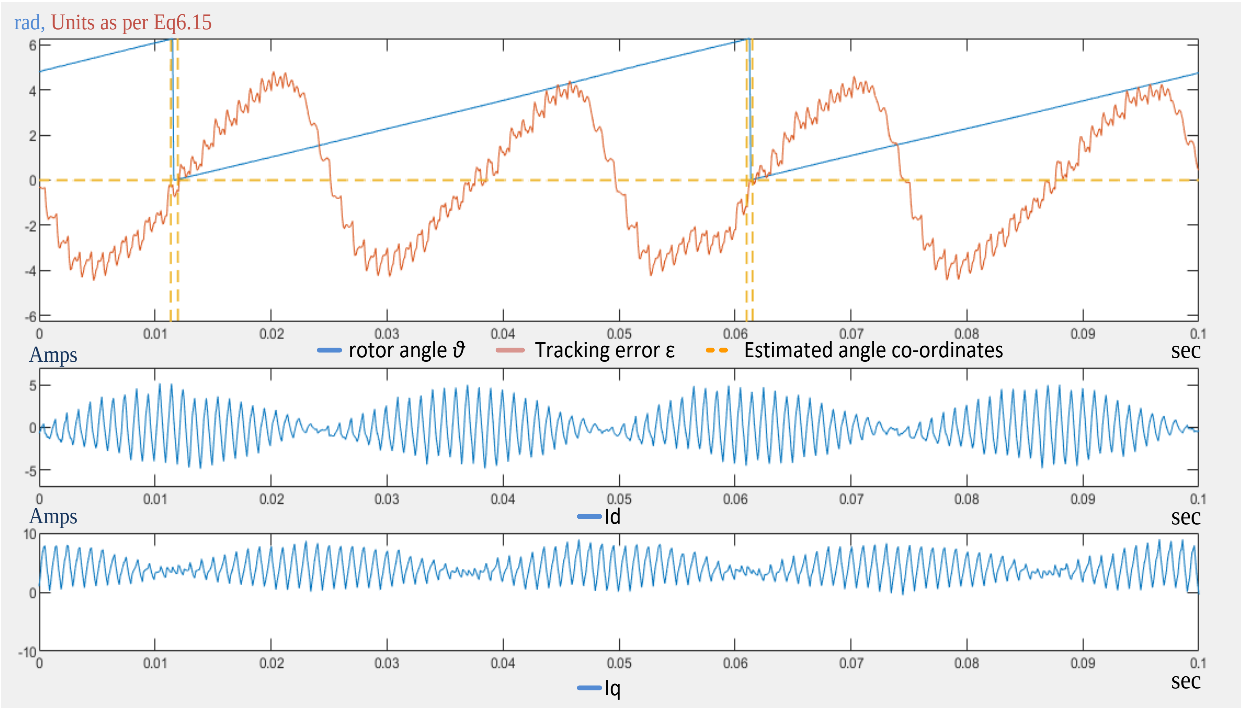


Figure 6.30 Hardware testing, saliency error when $\hat{\vartheta} = 0$ rad and the motor is spinning at 20Hz

Hardware testing scenario1 $\hat{\vartheta}= \pi/3$ rad and motor spinning at 20Hz

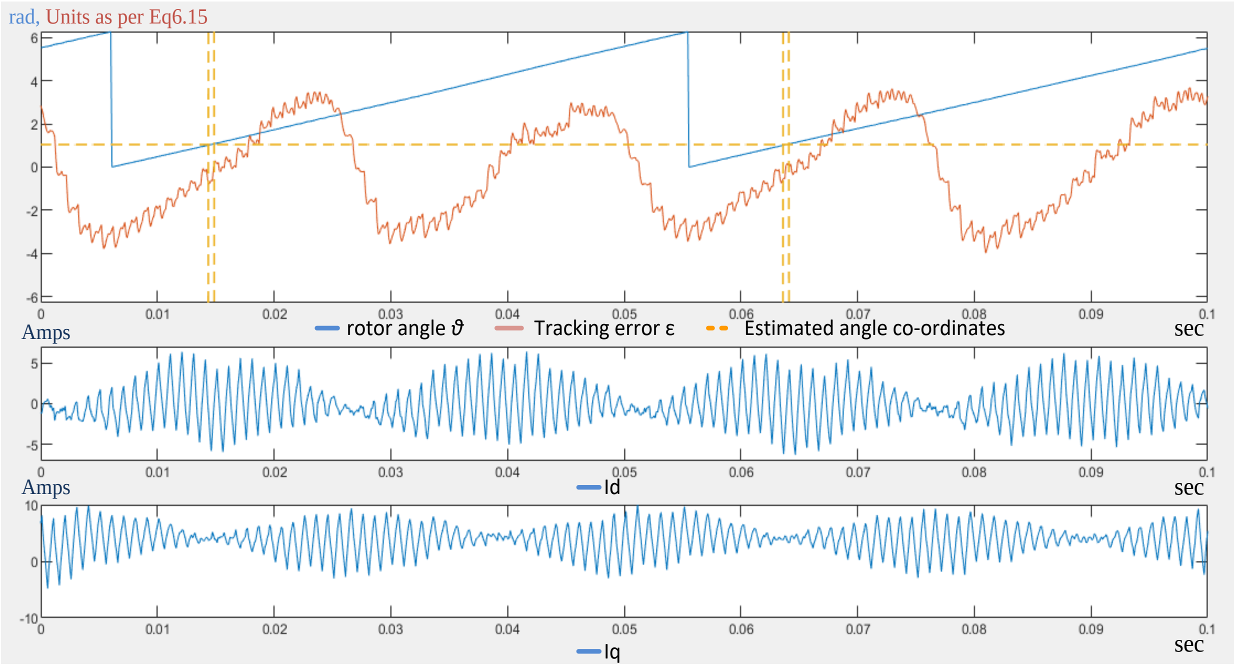


Figure 6.31 Hardware testing, saliency error when $\hat{\vartheta} = \pi/3$ rad and the motor is spinning at 20Hz

Hardware testing scenario1 $\hat{\vartheta}= 2\pi/3$ rad and motor spinning at 20Hz

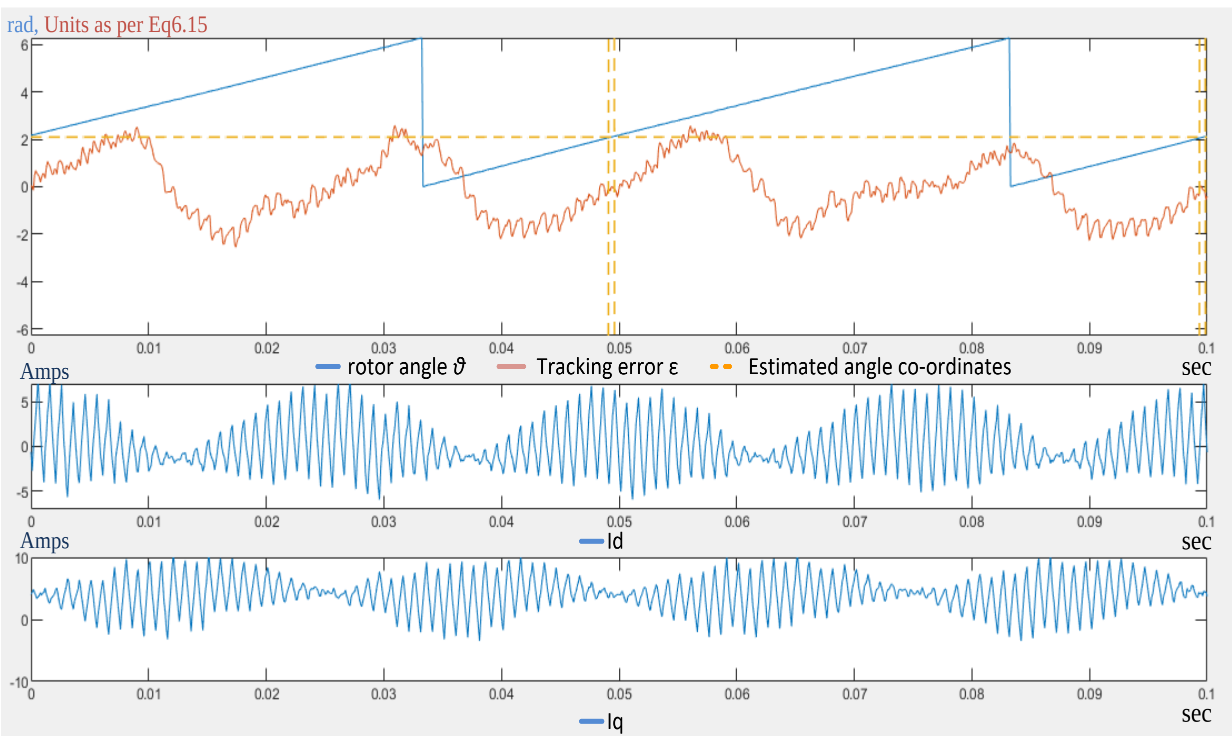


Figure 6.32 Hardware testing, saliency error when $\hat{\vartheta} = 2\pi/3$ rad and the motor is spinning at 20Hz

Hardware testing scenario1 $\hat{\vartheta}$= $3\pi/3$ rad and motor spinning at 20Hz

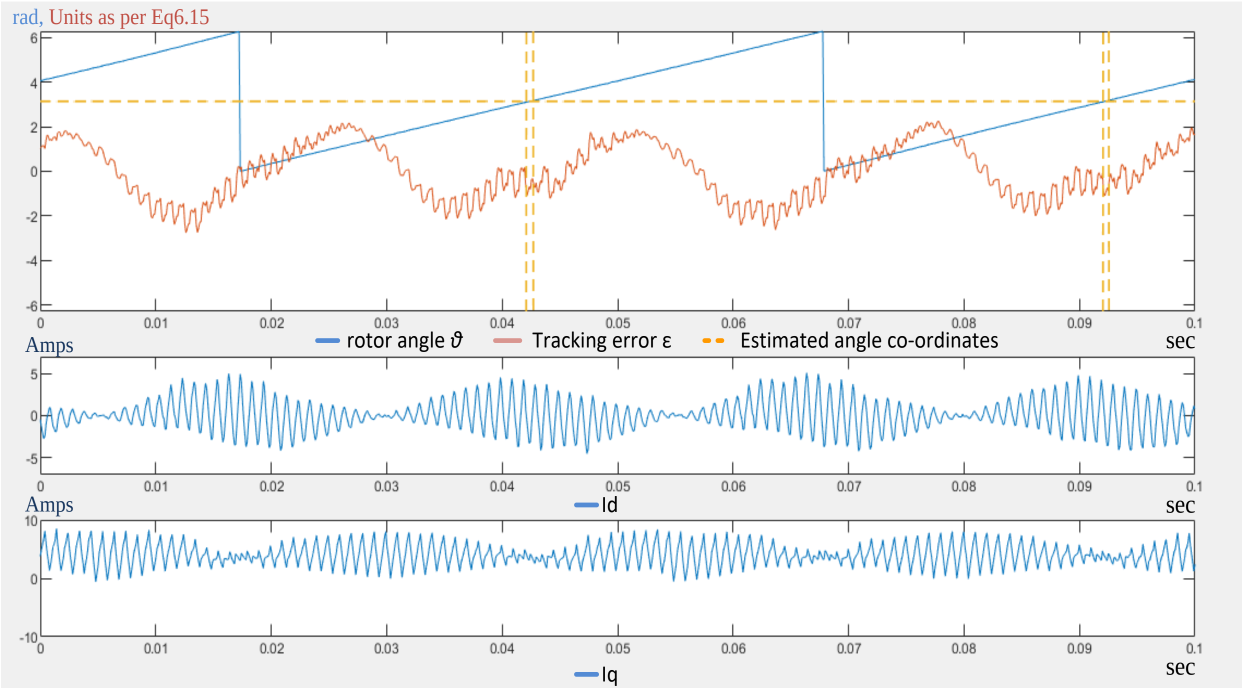


Figure 6.33 Hardware testing, saliency error when $\hat{\vartheta} = 3\pi/3$ rad and the motor is spinning at 20Hz

Hardware testing scenario1 $\hat{\vartheta}$= $4\pi/3$ rad and motor spinning at 20Hz

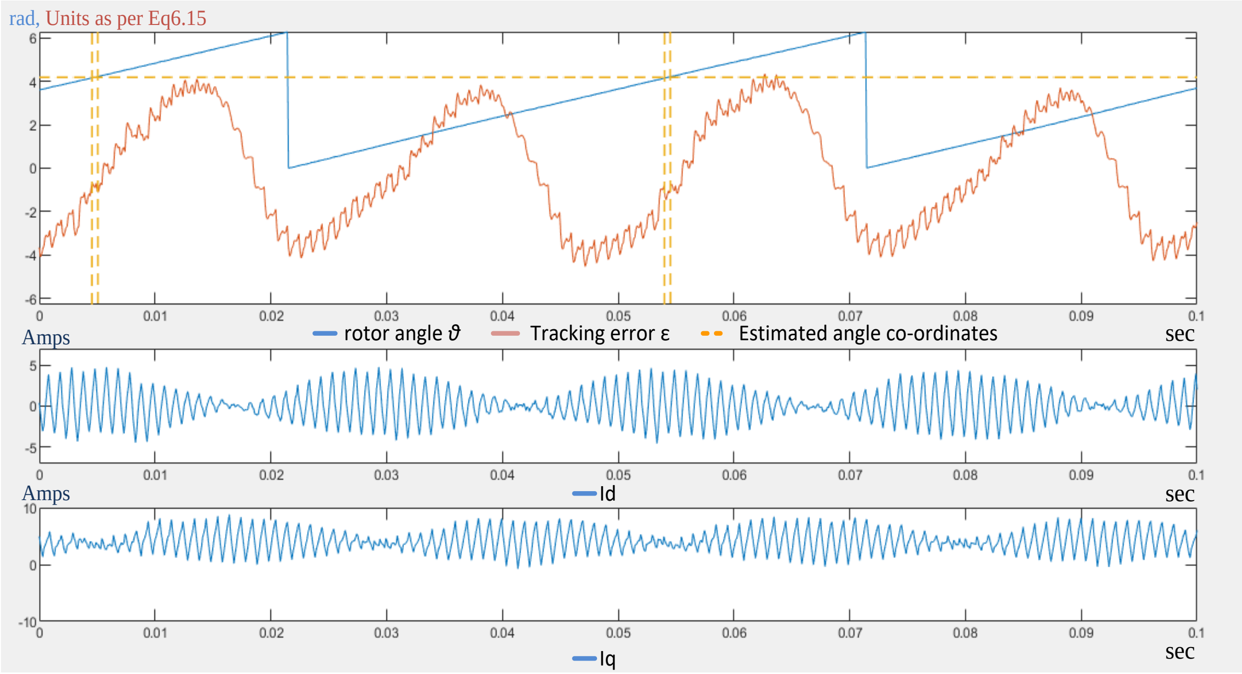


Figure 6.34 Hardware testing, saliency error when $\hat{\vartheta} = 4\pi/3$ rad and the motor is spinning at 20Hz

<u>Hardware testing scenario1 $\hat{\vartheta}$= $5\pi/3$ rad and motor spinning at 20Hz</u>

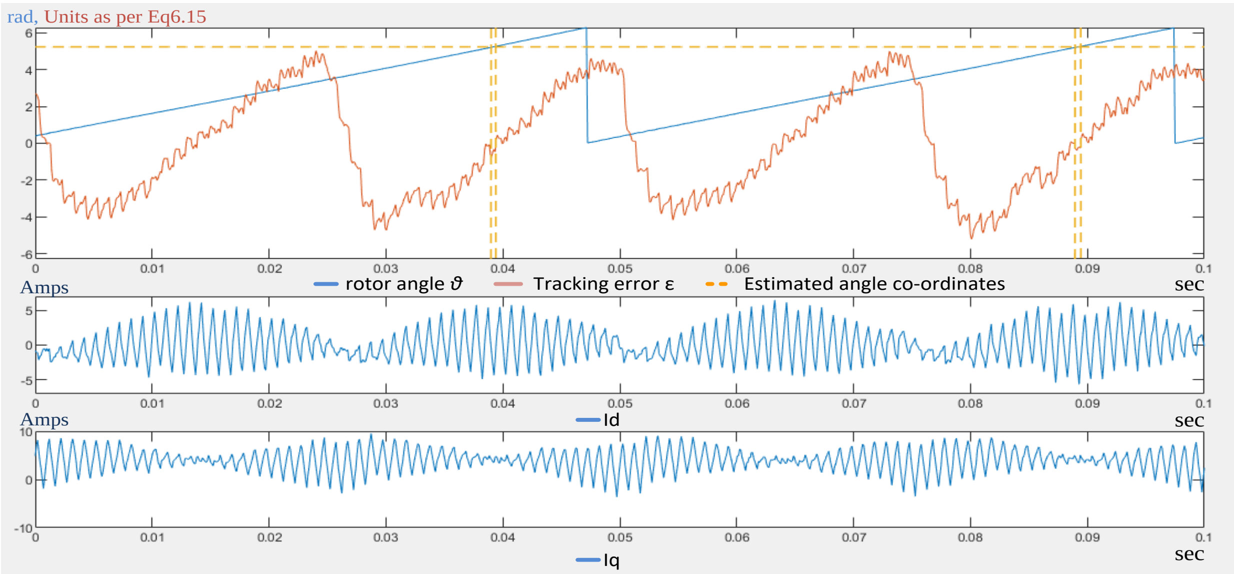


Figure 6.35 Hardware testing, saliency error when $\hat{\vartheta}$ $= 5\pi/3$ rad and the motor is spinning at 20Hz

Analysis of test results for scenario1:

- A sinusoidal like saliency error is observed having the value of 0 when $\Delta\vartheta \approx 0$, positive when $\Delta\vartheta$ is turning positive and negative when $\Delta\vartheta$ is turning negative (i.e. following a sinusoidal like waveform.
- When $\Delta\vartheta = 0$ all of the HFI voltage is deposited on the d axis as observed in the d axis current i_d. When $\Delta\vartheta \neq 0$ portion of this HFI voltage and specifically the resulting current can be seen on the q axis.
- The angle error signal still contains some HFI components. This is because applying a stronger LPF would introduce greater delay that would compromise the response to fast angular changes.
- The shape of the saliency error signal is not exactly sinusoidal. Simulink simulation Fig. 6.6. bottom (where L_d, L_q change as a function of current in d axis) and VHDL simulation Fig 6.14 have also shown similar distortions. Analysing the simulations, it can be seen that as the actual angular error increases, the current deposited to the d axis and the motor's saliency decrease causing the sensorless tracking error signal to follow a non-sinusoidal relation due to equation 6.15. In other words, if saliency was constant, the tracking error would be sinusoidal, however as the saliency decreases, the amplitude of the tracking error also decreases. Also, considering the very low levels of saliency, the ADC quantization and sampling noise also have an effect in these calculations.
- The saliency error in hardware testing of scenario1 was less sinusoidal than VHDL simulation that was in turn less sinusoidal than the Matlab modelling. The closer the algorithm is brought to the hardware in terms of simulation and testing (abstract to implementation level), the stronger the effect of system considerations such as sampling noise to the low saliency error.

6.9.2 Test Scenario 2

In this testing, the motor is spinning using the resolver and the saliency HFI method is attempting to track the rotor angle. The test results below are initially for motor speed of 15Hz (180 rpm). Note that the test conditions for Figures 6.36/6.37 are identical and simply illustrate the tracking over two captured windows. At 20Hz It was found that the angle tracker is slow resulting to less voltage being placed to the d axis that in turn reduced the saliency and the tracker becoming out of sync and unstable. The PI gains were therefore increased as shown in Table 6.3 and the results tracking at 20Hz are presented.

<u>Hardware testing scenario2 tracking motor angle spinning at 15Hz</u>

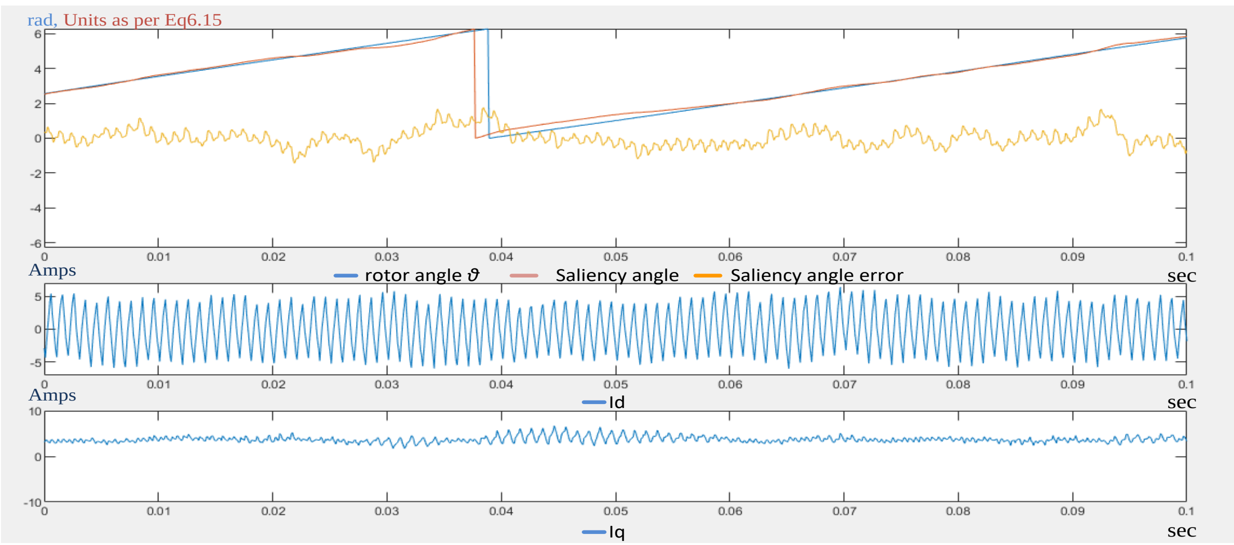


Figure 6.36 Hardware testing, tracking motor angle when spinning at 15Hz 1 of 2

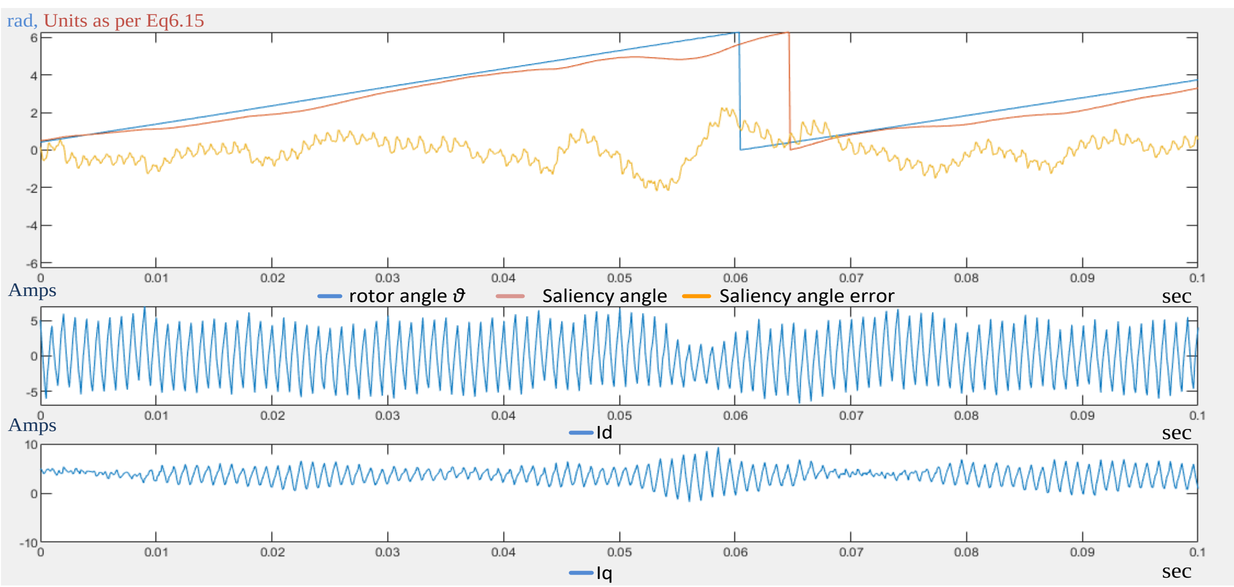


Figure 6.37 Hardware testing, tracking motor angle when spinning at 15Hz 2 of 2

<u>Hardware testing scenario2 tracking motor angle spinning at 20Hz</u>

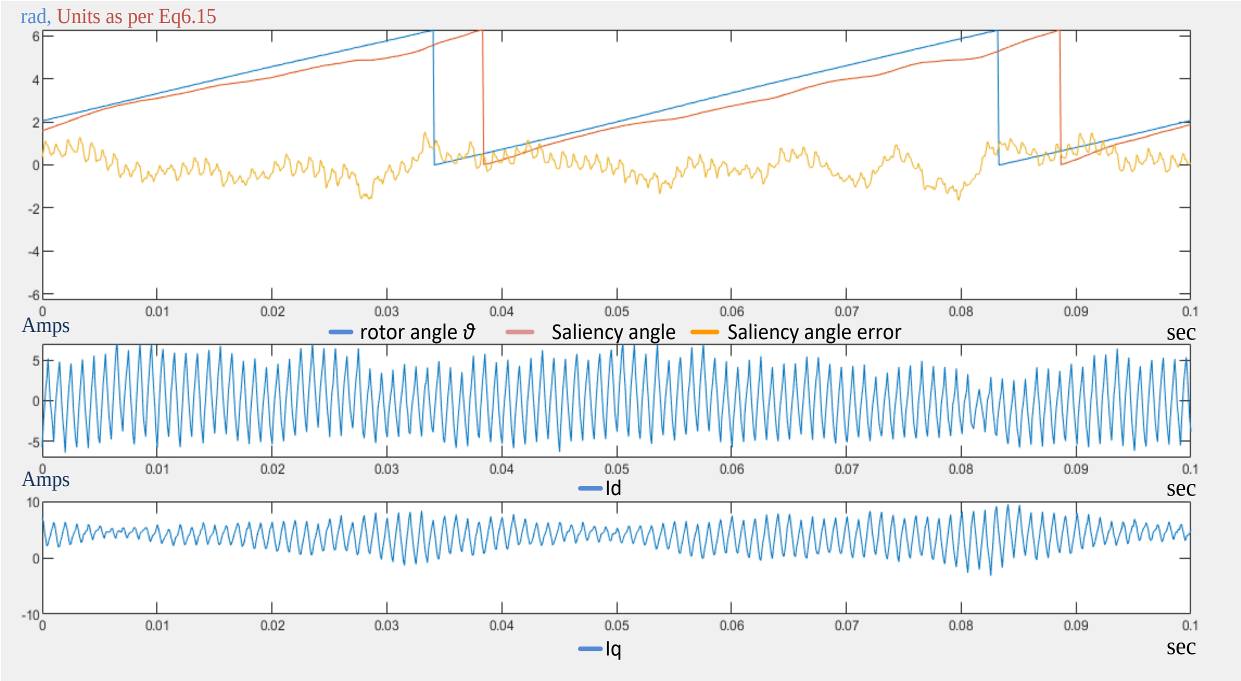


Figure 6.38 Hardware testing, tracking motor angle when spinning at 20Hz

Motor speed and test No	angle error max (deg)	%
15Hz, test 1/2 (Kp=Ki=0.5)	21.4	5.9%
15Hz, test 2/2 (Kp=Ki=0.5)	59.8	16.6%
20Hz (Kp=Ki=0.6)	61.8	17.1%

Table 6.3 Hardware testing scenario2 results analysis table

Analysis of test results for scenario2:

- The tracker is able to track the rotor angle while spinning at 15Hz. Tracking the rotor at 20Hz required an increase in the tracking PI gains. However increasing the PI gains sometimes resulted into the tracking loop becoming unstable and losing sync. Setting of Kp=Ki working for 15Hz or lower motor speed was found to be most stable in angle tracking. The saliency method is best suited for low speed as at higher speed the BEMF method illustrated in chapter 5 can take over so this limitation was found acceptable.
- As the angle error increases a portion of the HFI voltage is noticed to be deposited to the q axis (q axis current) which in turn results into the detection of saliency angle error that the PI tracker responds to by corrects the angle

6.10 Analysis of simulation and test results

The simulation/test plan intended to provide a framework that can link theoretical formulas with simulation and hardware testing on the Infineon DSP. Two scenarios were simulated and tested. Scenario1 was useful towards understanding and validating the angle tracking equation 6.15. Scenario2 intended to use this saliency angle error towards tracking a spinning motor. Simulation showed early on that the saliency and therefore tracking signal would be of relatively low amplitude and its shape distorted. Additionally during testing the below lessons learnt were also be noted:

- One mistake made in early testing was that HFI was applied at relatively low frequency of a few hundred Hz (500 Hz). This resulted into the HFI being within the bandwidth of the current control loop which caused the current PI controller to fight the HFI. The current control loop would observe the positive half of the HFI sinusoid feedback current and oppose it with a negative voltage demand and similarly for the negative part with a positive voltage demand. This was resolved by increasing the injection frequency of the sinusoid. One thing to note is that increasing the frequency from 500Hz to 1kHz also necessitated almost doubling the voltage of the HFI sinusoid so as to maintain the same feedback current of $\approx 5.2A$ and the desired saliency due to stator iron saturation.
- The BEMF observer method presented in Chapter 5 exhibited more ease of testing and more accuracy in terms of angle tracking versus the saliency method. This is due to the magnitude of the physical parameters of each method. The Back EMF voltage measures in tens of Volts over the motor's speed range (see Figure 5.2) but the saliency of the HEMAS motor is substantially small ($\frac{L_q}{L_d} = \frac{1.158}{1.069} = 1.083$) even when 5.2A is invested to establish saturation. This low saliency and low saliency tracking signal is strongly affected by system considerations such as ADC sampling noise, PWM switching effects and Dead Time Distortion.
- One overall comment was that while the saliency method was shown to work one could reason that 5.2A is excessive for an application merely towards tracking the rotor angle. This chapter though aimed to provide the background simulation and implementation detail of the method which could be more easily applied to a slightly more salient motor. An alternative option if saliency levels remain low is to use an open loop method to accelerate the motor and when the speed is high enough switch to the BEMF method [80]. This combination of methods has been applied on a separate project by the author and works well for very low saliency SPMSMs. Note that open loop acceleration involves applying a rotating stator magnetic field without direct knowledge of the rotor angle and the motor spinning due to the field coupling between stator and rotor.

6.11 A novel saliency rotor tracking method for SPMSMs

Towards the end of the research effort placed on this chapter, a novel saliency method was conceived. One key side effect of stator iron saturation is that a HF voltage signal (sinusoid or square wave) that is injected to the d-axis, will result into an asymmetric current i.e. with the positive half having higher amplitude than its negative half. This is due to L_d having reduced value at the saturation point creating higher positive current spike versus its negative half. An algorithm is therefore proposed scanning the whole 360 degrees range for the initial rotor position identifying the point where the HF positive current is at its peak versus its negative half. While the motor is spinning a similar method is proposed injecting HF voltage over a smaller angular range centered at the last known position of the rotor and checking the positions on each side of the last known position taking under consideration the maximum possible speed of the motor. The initial and continuous rotor tracking methods are shown diagrammatically in Figure 6.39.

One key advantage of the proposed method is that a Low Pass Filter is not needed as would be the case for the saliency method shown of Figure 6.2. The lack of need for a LPF means that the proposed tracking method can be faster. This is because a Low Pass Filter introduces a delay in the calculation until the input data propagates to the output with this delay being proportional to time constant of the filter. In order to accommodate for possible DC offset of the current feedback the average current and peak to peak values are also proposed to be monitored.

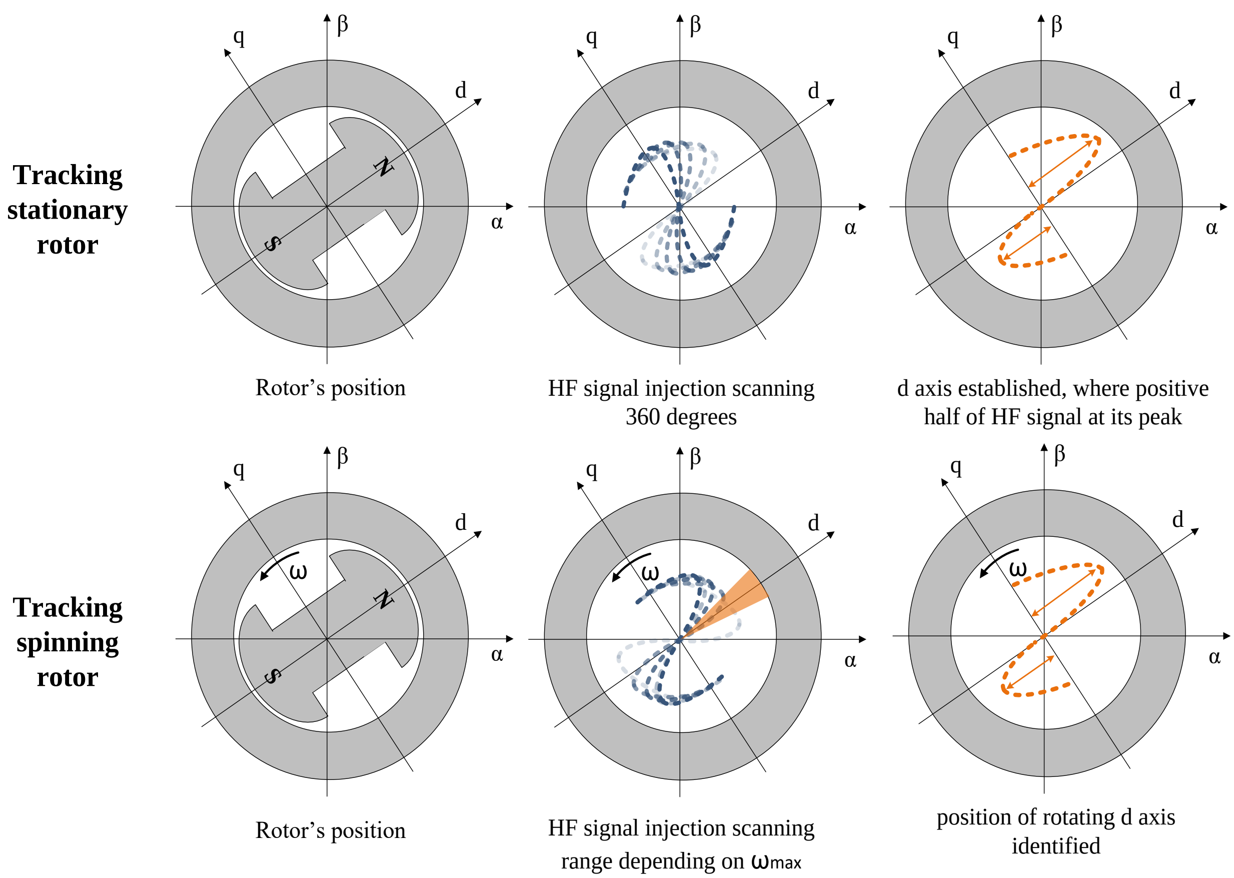


Figure 6.39 Proposed novel saliency algorithm

6.12 Conclusions

A theoretical background on saliency and HFI methods was first provided in this chapter aiming to introduce the reader to the domain of HFI. A saliency method was then proposed taking under consideration the characteristics of the HEMAS motor. Aiming to establish a comparative link between theory and implementation, simulation at multiple levels of abstraction regulated by a simulation plan was performed. Implementation for the targeted Infineon DSP and hardware testing on the HEMAS platform followed, completing a cycle from understanding the concept, following the theory down to hardware testing.

Chapter 7 PRHFI and Acoustic Noise

7.1 Introduction

High Frequency Injection (HFI) sensorless methods involve the injection of a high frequency voltage to a salient motor and the estimation of the rotor angle by means of analyzing the high frequency feedback currents. One limitation of this injection method is that the generated HF currents create torque pulsations that in turn cause vibrations within the motor that propagate into the air in the form of acoustic noise [62], [63]. As will be demonstrated in this chapter, the predominant portion of the acoustic noise from a constant frequency HFI is tonal and specifically at the frequency of the injection. Research has shown however that the human ear is more susceptible to pure tones rather than a wide frequency spectrum of equivalent amplitude [65]. An area of research in the past years aiming to reduce the perception of acoustic noise has been to vary the HFI frequency in a pseudo random manner [62], [63]. The PRHFI algorithm presented in this chapter, differs with respect to [62] and [63], firstly because a mixture of sinusoidal and square wave injection is proposed and secondly due to the amplitude compensation mechanism for PWM frequency limitations also presented in the chapter. These two novel aspects of the algorithm are used to improve acoustics of PRHFI and allow a wider acoustic frequency spectrum of the method. This chapter begins with an analysis of acoustic noise from HFI and then presents a variant of the Pseudo Random High Frequency Injection (PRHFI) algorithm detailing the proposed method. The analysis, simulation and test results of the method when applied to the HEMAS platform are then presented.

7.2 The mechanism of sound generation from HFI

A useful qualification before attempting to modify the acoustics from HFI is to understand the mechanism of sound generation from voltage injection. When high frequency voltage is applied to a motor's phase it creates HF currents that produce torque pulsations and vibrations within the motor. In order to characterise the acoustic generation process in more detail it is necessary to revisit the equivalent mathematical model of the motor of interest. In the case of the HEMAS platform the model of a PMSM in the synchronous frame is shown in the equation below:

$$\begin{bmatrix} V_d \\ V_q \end{bmatrix} = \begin{bmatrix} R_s i_d + \frac{d}{dt}(L_d i_d + \Psi_f) - \omega_r L_q i_q \\ R_s i_q + \frac{d}{dt} L_q i_q + \omega_r (L_d i_d + \Psi_f) \end{bmatrix} \qquad \text{Eq 7.1}$$

where V_d, V_q are stator voltages in the synchronous frame, R_s is the stator resistance, i_d and i_q are stator currents in d and q axis, ω_r is the rotor's electrical angular speed, L_d is the d axis inductance, L_q is the q axis inductance, and Ψ_f is the rotor flux.

Assuming the HFI method is applied to the estimated d axis and an estimation angular error θ_{err} exists, then a portion of the HF voltage is deposited to the d axis and the remaining portion is deposited to the q axis as per equations below:

$$V_{d\_HF} = V_{carrier} * (\cos(\omega_c t)) * cos\theta_{err} \quad \text{Eq 7.2}$$

$$V_{q\_HF} = V_{carrier} * (\cos(\omega_c t)) * sin\theta_{err} \quad \text{Eq 7.3}$$

where V_{d_HF}, V_{q_HF} is the high frequency voltage deposited to the motor's d and q axis, $V_{carrier}$ is the amplitude of the High Frequency carrier, ω_c is the frequency of the HF voltage, and θ_{err} is the angular error between actual and estimated angle.

Assuming that the voltage drop due to resistance $R_s * i_d$, $R_s * i_q$ and the Back EMF components (proportional to ω_r) are negligible relative to the voltage drop due to the current oscillation ($\frac{di_d}{dt}$, $\frac{di_q}{dt}$) equation 7.1 becomes:

$$\begin{bmatrix} V_{d\_HF} \\ V_{q\_HF} \end{bmatrix} = \begin{bmatrix} V_{carrier} * (cos(\omega_c t)) * cos\theta_{err} \\ V_{carrier} * (cos(\omega_c t)) * sin\theta_{err} \end{bmatrix} \approx \begin{bmatrix} L_d \frac{d}{dt}(i_d) \\ Lq * \frac{d}{dt} i_q \end{bmatrix} \quad \text{Eq 7.4}$$

Based on equations 7.2, 7.3, 7.4 if a sinusoidal High Frequency voltage V_{d_HF}, V_{q_HF} is applied to the d and q axis, it will create sinusoidal currents of the same frequency in the two synchronous axis frames. In order to calculate the resultant torque from the HF currents the torque formula for a PMSM is shown below:

$$T_{PMSM} = (3/2) * (P/2) * (\lambda_m * i_q + \left(L_d - L_q\right) * i_d * i_q) \quad \text{Eq 7.5}$$

where P is the number of poles, λ_m is the permanent magnet flux, L_d, L_q is the d and q axis motor inductance and i_d, i_q is the d and q axis motor current.

Taking under consideration equations 7.4 and 7.5 it is illustrated analytically that a sinusoidal HF voltage of n kHz, results into HF currents (i_d, i_q) and HF torque that are also sinusoidal and centred at the same frequency. Considering the theory on acoustics introduced in chapter 3, the vibration of the motor's mechanical infrastructure at n kHz creates vibration of the air molecules surrounding the machine at the same frequency that in turn causes a travelling sound also centred at frequency of n kHz. This analytical conclusion on the frequency of the acoustic noise from HFI is validated by means of hardware testing in the experimental section of this chapter.

Note: although it is claimed above that HFI of constant frequency creates acoustic noise at the injection frequency, as will be demonstrated experimentally, along with this fundamental frequency what is also observed are harmonics of this frequency.

7.3 The proposed Pseudo Random High Frequency Injection algorithm

Having identified analytically the correlation between HFI and acoustics, it is of interest to attempt to modify the acoustic noise from HFI to reduce its perception by the receiver. This section therefore proposes an algorithm that varies the injection frequency and as a result the frequency spectrum of the generated acoustic noise. Previous research work in the area of pseudo random frequency injection exists specifically [63] and [62]. However one unique advancement provided by the algorithm detailed in this chapter is the dynamic transition between sinusoidal and square waveform injection aiming to provide an acoustically smoother and wider frequency spectrum of frequencies. Additionally, in order to compensate limitations imposed by the selected PWM frequency in injection and acoustic frequency spectrum, an amplitude compensation method is also proposed in the following section.

7.3.1 Description of the algorithm

Injection frequency variation: the algorithm's first key responsibility is changing the frequency of the injected voltage in a Pseudo Random manner. This variation is achieved by means of a commonly used method of shift registers that results into a continuous variation of the injection frequency to what an external observer would perceive as random. In practice this pseudo randomness produces a pattern that is repetitive over time but its period is long enough for the human ear not to recognize as periodic.

Injection amplitude variation: adhering to equation 6.1 and in order to establish the same HF current as the injection frequency increases, a proportionally higher HF voltage amplitude is needed. Maintaining the same levels of HF currents for different frequencies is key in order to have constant sensorless calculation accuracy and establishing the same amount of saturation for an SPMSM. A key task of the algorithm is therefore to vary the amplitude of the HF voltage as a function of the injection frequency.

Upper and lower injection frequency range: one consideration of the PRHFI algorithm is the selected frequency range of the HFI. As discussed in chapter 6, HFI should be set high enough so that it does not interact with the current control loop and the speed time constants of the motor drive. The algorithm therefore allows the user to set the upper and lower frequency range of the PRHFI in a user-friendly manner at run time while the DSP code is running.

Dynamic sinusoidal/square waveform voltage variation: an implementation parameter that affects a HFI algorithm and its injection frequency range is the PWM frequency. The default PWM frequency for the HEMAS controller is 10kHz. If sinusoidal injection was to be performed at 1kHz this would allow for 10 HF voltage points in the voltage control loop. At higher injection frequency the number of voltage points is reduced and the injection frequency is less distinct. In order to enable higher injection frequency, the proposed method automatically switches from sinusoidal injection to square wave when the pseudo random frequency is above a predefined threshold (in this case 2kHz). The transition point of 2kHz was selected as it provided sufficient number of sinusoidal voltage injection points (worst case of 5 when approximating 2kHz) allowing the reconstruction of audio at the specified frequency. Taking under consideration that the human ear cannot perceive higher frequencies equally well [81], the question that comes about is why should one use both sinusoidal and square wave injection and why not simply use only square wave injection at higher frequency assuming there is no impact on wild life that is able to perceive such frequencies. Based on testing performed in this research and some analysis on acoustics [82], the square wave voltage injection and resultant triangular current waves create acoustic noise that sounds more spiky, abrupt, and irritant to the ear rather than a sinusoid that sounds smoother. The proposed algorithm is therefore aiming to join the best features of frequency and waveform shape injection. The use of square wave injection enables the frequency spectrum to reach higher frequencies that is less audible by a human ear, while at the lower end of the injection spectrum sinusoidal injection is used that sounds smoother. One additional consideration in terms of square wave injection relates to the amplitude of the injected voltage. To establish the same injected input power between a sinusoidal and square wave of the same frequency, the square wave voltage needs to be of lower amplitude than the sinusoidal wave. This relates to rms theory specifically if a voltage $V_{(t)}$ is applied to a passive load R, the power dissipated on this load is $V_{(t)}^2/R$. To calculate the RMS of a signal in general, Equation 7.6 holds

$$V_{rms} = \sqrt{\frac{1}{t}\int_0^t V_{(t)}^2 dt} \quad \text{Eq 7.6}$$

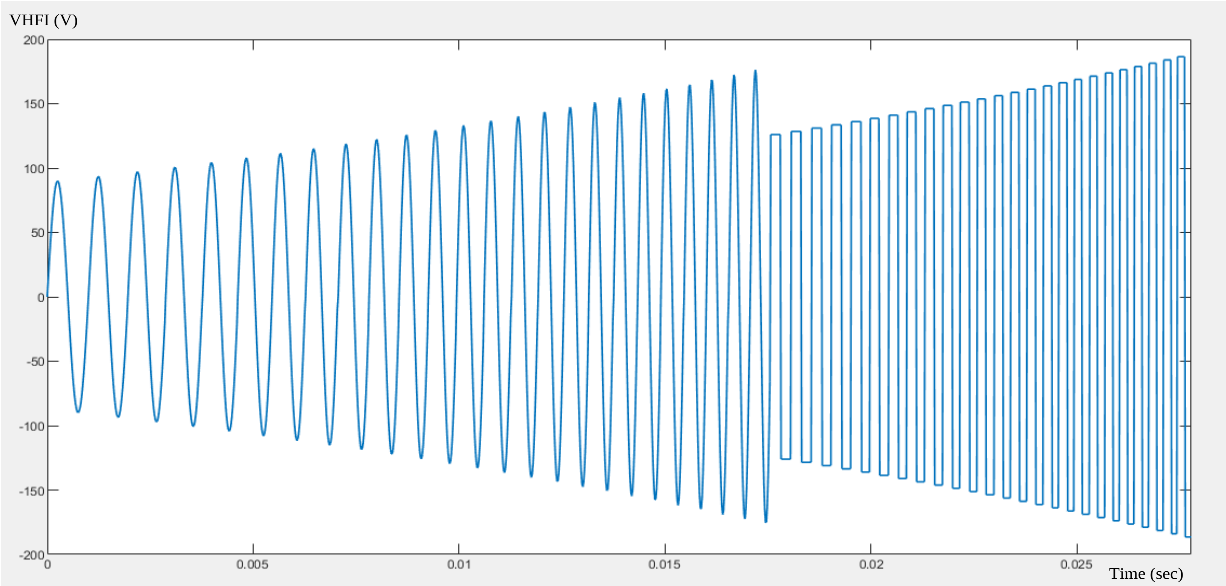
where V_{rms} is the Root Mean Square voltage i.e. the DC voltage that would produce an equivalent power dissipation of $V_{(t)}$.

Assuming $V_{(t)}$ is sinusoidal Equation 7.6 and performing integration from 0 to π equation 7.6 becomes:

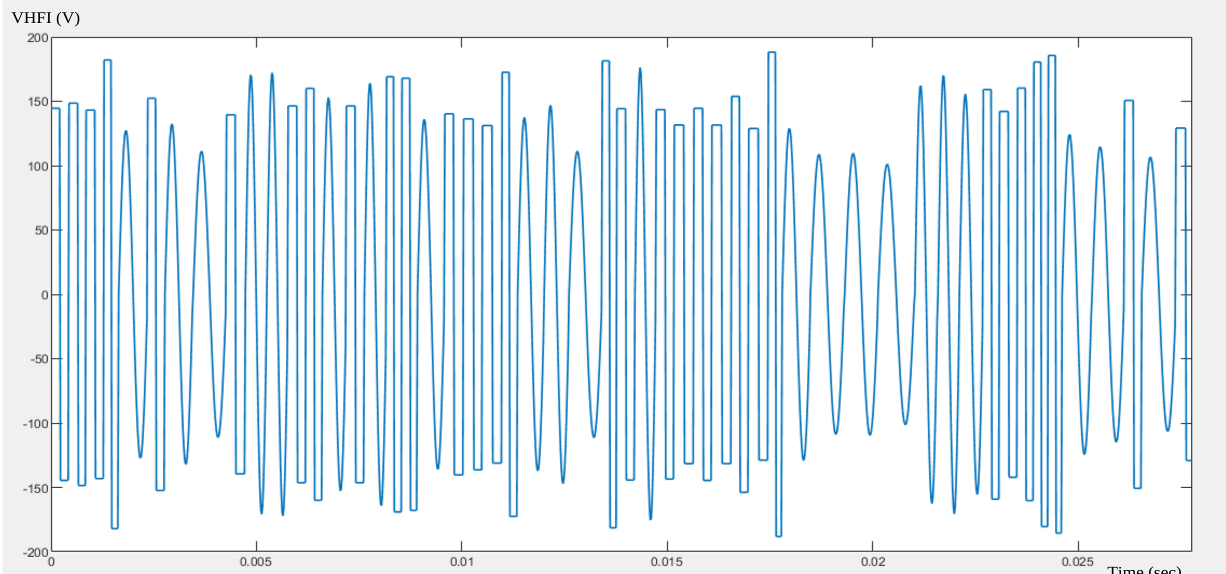
$$V_{rms} = \sqrt{\frac{1}{\pi}\int_0^\pi sin_{(t)}^2 dt} = \sqrt{\frac{1}{\pi} * \frac{\pi}{2}} = \sqrt{\frac{1}{2}} \approx 0.707 \quad \text{Eq 7.7}$$

So as per equation 7.7, the amplitude of the square wave would need to be 0.707 of the amplitude of the square wave to have equivalent input power injection. However, considering the difference in the phase current waveform shapes (sinusoidal versus triangular) the ratio of 0.707 would generate slightly higher current peaks for square wave injection relative to sinusoidal. To ensure the peak of the currents are the same between the two waveforms the sinusoidal to square voltage ratio is therefore reduced from 0.707 to 0.7 instead.

The PRHFI algorithm is presented visually in Figure 7.1.a illustrating the voltage pattern of the injection algorithm if the injection frequency was to increase over time. This figure aims to illustrate how the amplitude of the sinusoidal injection increases with frequency and reduces when changing from sinusoidal to square wave. Figure 7.1.b shows how the actual PRHFI algorithm functions where the frequency changes in a pseudo random manner therefore introducing waveforms of different amplitude, frequency and shape over time.



7.1.a HFI voltage with varying frequency incrementing from 1kHz to 3kHz, illustrating that when f_HFI > 2.0kHz waveform is square



7.1.b HFI voltage with Pseudo Randomly changing frequency from 1kHz to 3kHz, square when f_HFI > 2.0kHz

Figure 7.1 Pseudo Random High Frequency Injection algorithm

*Note that a Matlab model *.m file was written to illustrate the Pseudo Random High Frequency Injection algorithm when injection frequency varies from 1kHz to 3kHz as shown in Figure 7.1 please refer to Appendix B1.8 of this thesis to obtain the Matlab source code.*

7.3.2 Sinusoidal to square waveform feature analysis

Previous section supported the key feature of the proposed algorithm enabling continuous sinusoidal/square transitions claiming that the noise from square wave injection is more abrupt, spiky and irritant to the human ear. This section attempts to justify this statement.

A Matlab file was written to create two sets of waveforms, one representing current/torque/acoustics from sinusoidal injection (Figure 7.2 Top) and one from square wave injection (Figure 7.2 Bottom). The reader can either load the written Matlab script provided and evaluate the generated acoustics or click to the audio links uploaded online.

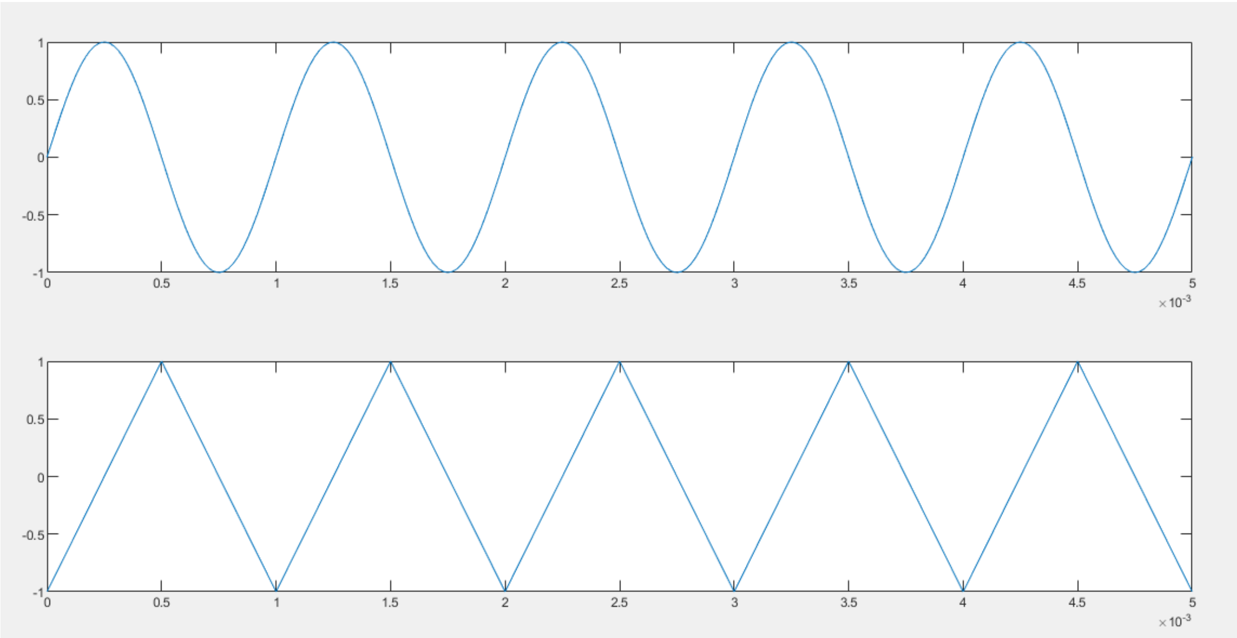


Figure 7.2 Pseudo Random High Frequency Injection algorithm

*Note that a Matlab model *.m file was written to illustrate the visual analysis and acoustic effects of sinusoidal versus square wave injection as shown in Figure 7.2 please refer to Appendix B1.9 of this thesis to obtain the Matlab source code.*

Audio link 7.1: Matlab acoustics, sinusoidal injection:	https://youtu.be/kc5FqaOmWv8
Audio link 7.2: Matlab acoustics, square injection:	https://youtu.be/yAr_ykOfEGU

One physical observation when listening to the resultant noise from the two Matlab waveforms is that the sinusoidal noise sounds substantially smoother and less spiky than the triangular waveform. This is a sensible observation taking under consideration the shape of the two waveforms in Figure 7.2. Taking under consideration the acoustic difference between the two waveforms it is therefore supported analytically that there are benefits from using sinusoidal injection at lower injection —a feature of the proposed algorithm.

7.4 Implementation of the PRHFI algorithm

The original C code for the HEMAS controller was extended with the functions listed in Appendix B.2.2.2 and specifically function:

```
void PRHFI_gen(int low_freq, int high_freq, float amplitude_d)
```

Function `PRHFI_gen` takes as inputs user defined parameters `low_freq`, `high_freq` and `amplitude_d`. The first of the three parameter parameters defines the lower frequency while the second parameter defines the upper frequency of the pseudo random high frequency injection spectrum. The third parameter `amplitude_d` defines the amplitude of the injection while the injection is at low frequency. There are calculations within the function to change the injection amplitude as a function of the frequency and the switching between sinusoidal and square wave injection.

7.5 Test plan for the PRHFI algorithm

Aiming to evaluate the acoustic improvements from the proposed PRHFI algorithm, the hardware testing in this chapter is proposed to be composed of the below steps:

- Capture DSP data (voltage injection, motor current feedback Id/Iq) and acoustic data from conventional constant frequency HFI. This testing is to cover both sinusoidal and square wave injection and include a number of injection frequencies.
- Capture DSP data (voltage injection, motor current feedback Id/Iq) and acoustic data from the proposed Pseudo Random High Frequency Injection.

Following the above two sets of testing, a comparison is to be performed characterising the acoustic improvements obtained from the proposed PRHFI algorithm. While an analytical comparison of DSP and acoustic data is useful, the evaluation of the proposed method would not be considered complete if the user did not experience the acoustic improvement. Web links of the captured audio are therefore also provided to allow independent evaluation of the acoustic results.

7.6 Hardware testing

This section presents the DSP and acoustic testing results when constant and Pseudo Random High Frequency Injection (PRHFI) methods are applied to the HEMAS platform. Presenting the test data sequentially — constant and PRHFI — aims to make the analytical and acoustic data available for comparison therefore allowing the reader to reach a conclusion whether there is a gain towards the proposed method. The links to the audio data are provided in the end of this section.

The format of the DSP and acoustic data presentation and its underlying purpose is shown in Figure 7.3.

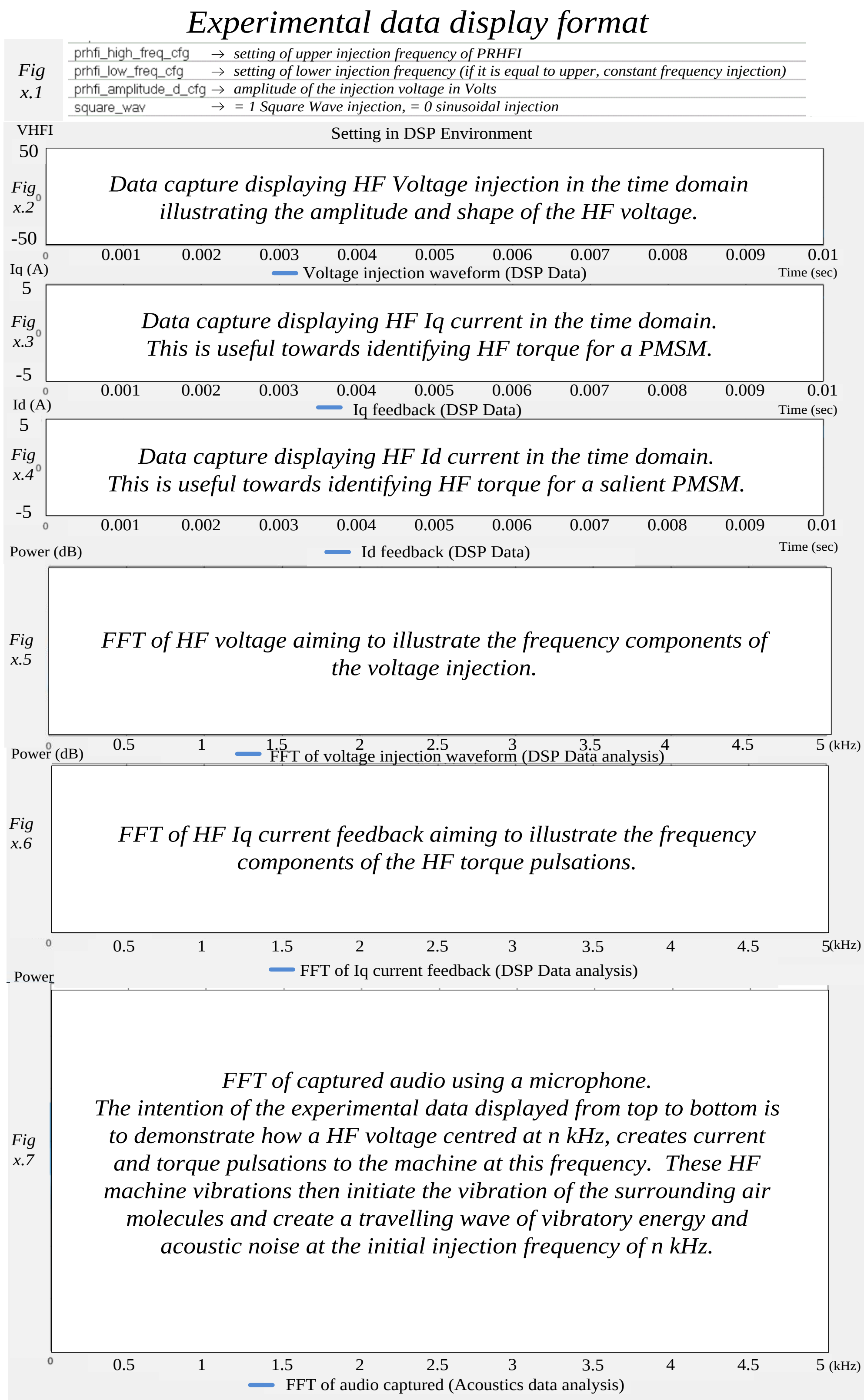


Figure 7.3 DSP and acoustic data display format

7.6.1 DSP data and Acoustics from constant frequency HFI

Constant frequency HFI - 1kHz Sinusoidal, DSP and Audio Data

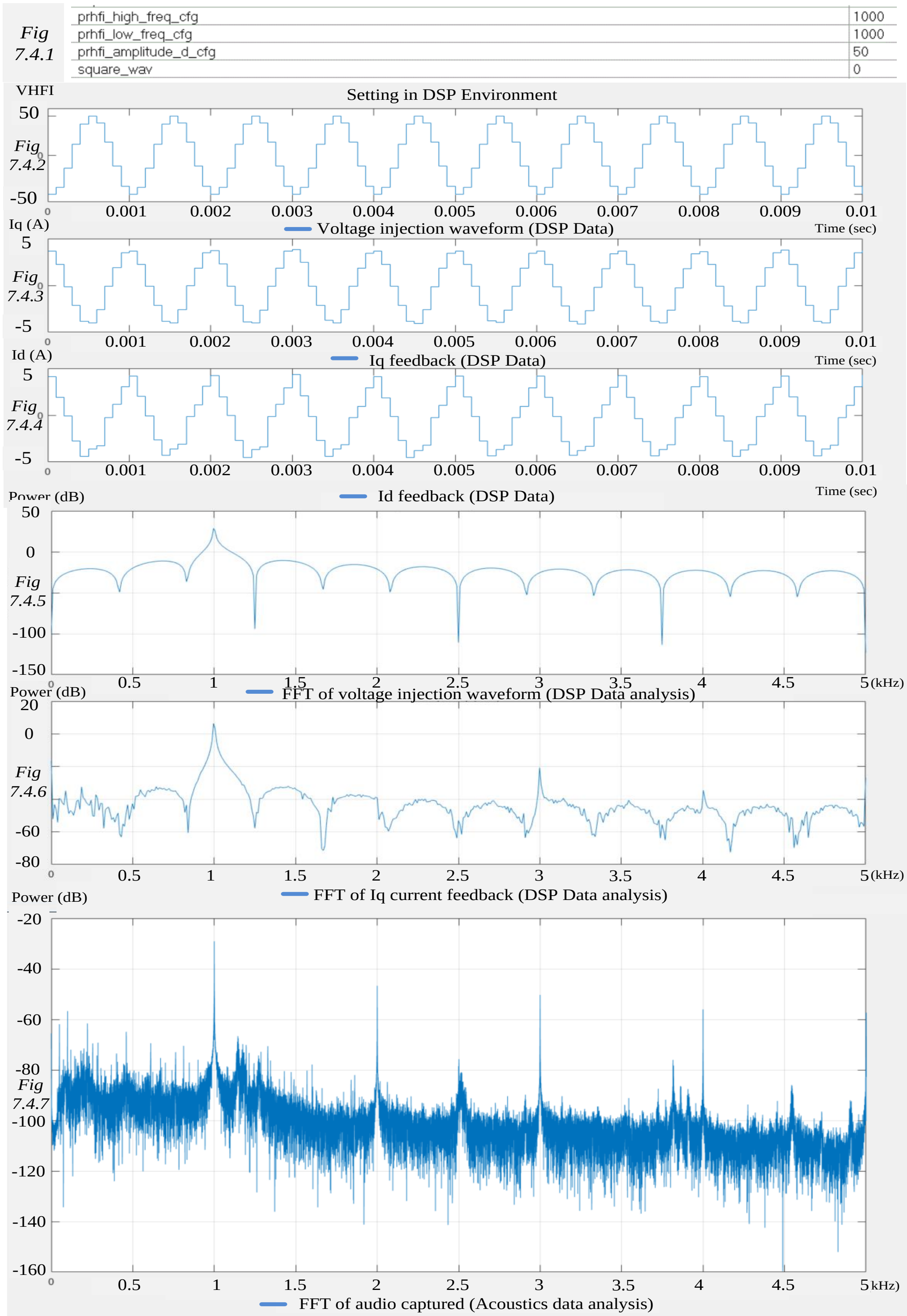


Figure 7.4 DSP logged data and acoustic noise capturing from sinusoidal HFI at 1kHz on the HEMAS motor

Viewing the test data illustrated in Figure 7.4, the first observation that can be made is that 1kHz HF voltage injection to the HEMAS platform results into 1kHz HF current along with its harmonics. Harmonics exist in the motor currents and acoustic FFT firstly because the PWM setting at 10kHz cannot construct a perfect 1kHz sinusoid and the voltage steps result into the distortion of the motor phase current waveform seen as harmonic content. Secondly harmonics exist due to non-linear switching devices within the inverter. Thirdly the harmonics observed in the acoustic FFT are also due to audio digitization process and sampling noise. The physical effect of the HF currents from this injection is that the rotor and stator experience torque pulsations and vibrations at the injection frequency. This vibration propagating throughout the metal infrastructure of the machine in turn excites the surrounding air molecules at the same frequency initiating their vibration. A travelling wave of vibratory energy in the air is thus initiated as per acoustic theory of Chapter 3 that reaches the receiver (ears and microphone). Via this electromechanical acoustic transfer mechanism, the 1kHz voltage injection resulted into acoustic noise centered at 1kHz plus its harmonics. This correlation is also shown in Figure 7.5 highlighted by the orange square. One secondary observation is that the acoustic FFT looks more accurate than the DSP data FFT with the latter appearing to suffer from spectral leakage. This is because the former is sampled at 48kHz with many samples constructing a long time window while the latter is limited to the sampling of the DSP at 10kHz sampling rate and to a relatively smaller number of samples due to hardware limitations. Finally when comparing current and acoustics FFT some additional harmonics appear in the acoustics. These additional harmonics are due to audio sampling noise that results into mild distortion of the sinusoidal acoustic wave appearing as harmonics in the FFT. Note that the analysis for the subsequent injection frequencies is very similar to the 1kHz injection and is presented in section 7.7 of this thesis.

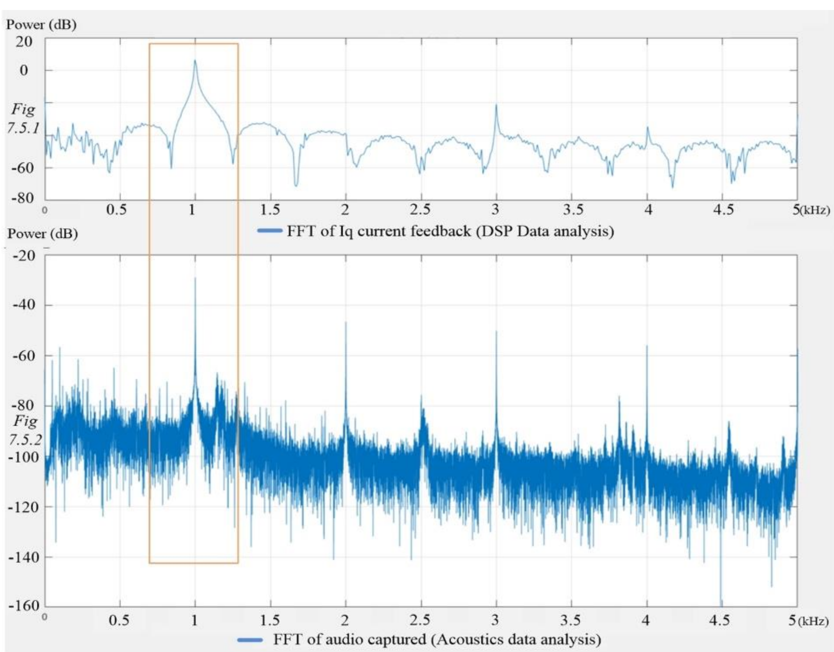


Figure 7.5 Analysis of data from sinusoidal HFI at 1kHz on the HEMAS motor

Constant frequency HFI – 1.4kHz Sinusoidal, DSP and Audio Data

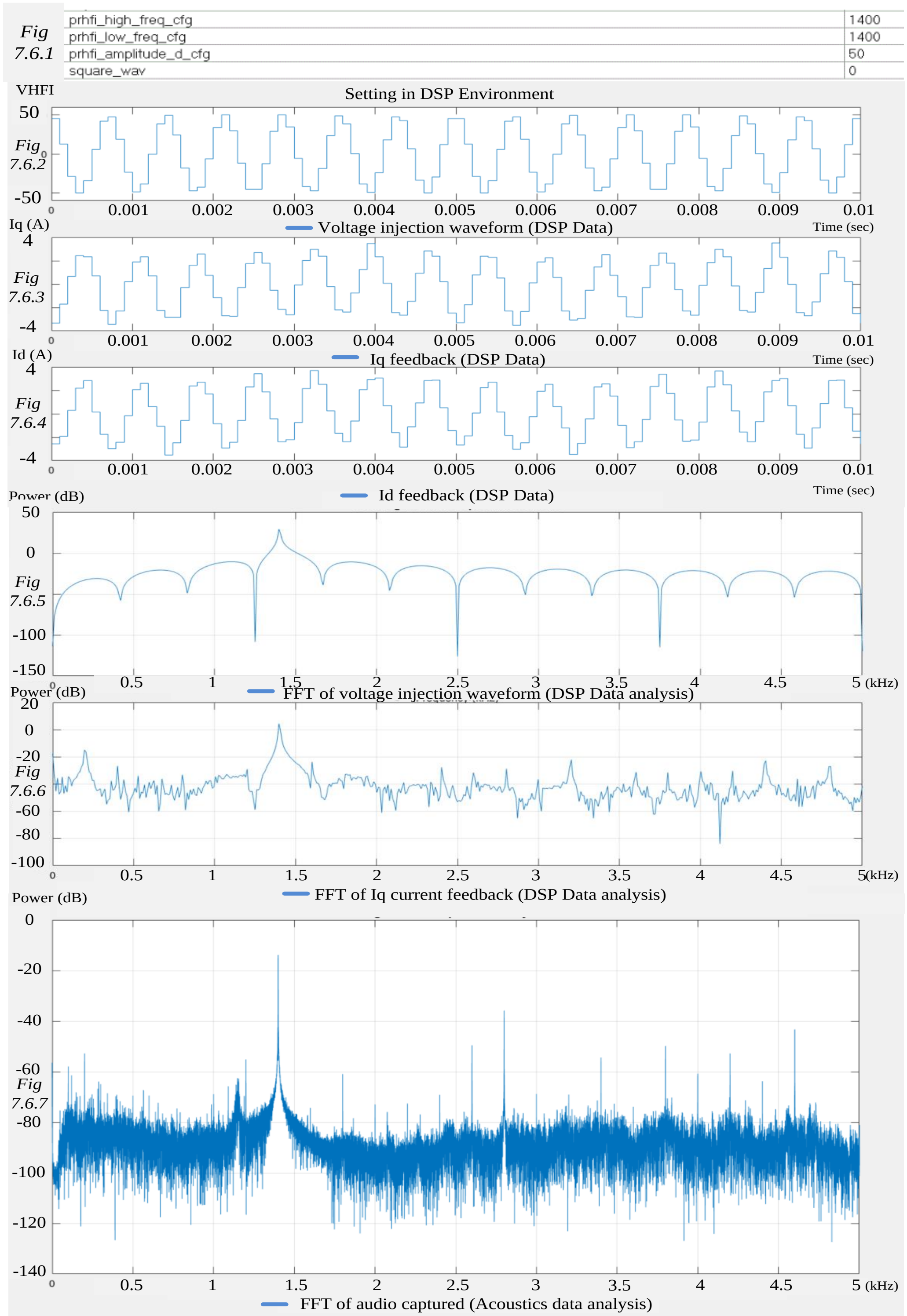


Figure 7.6 DSP logged data and acoustic noise capturing from sinusoidal HFI at 1.4kHz on the HEMAS motor

Note that analysis of the above and subsequent test results can be found in section 7.7 of this thesis.

Constant frequency HFI – 1.5kHz Sinusoidal, DSP and Audio Data

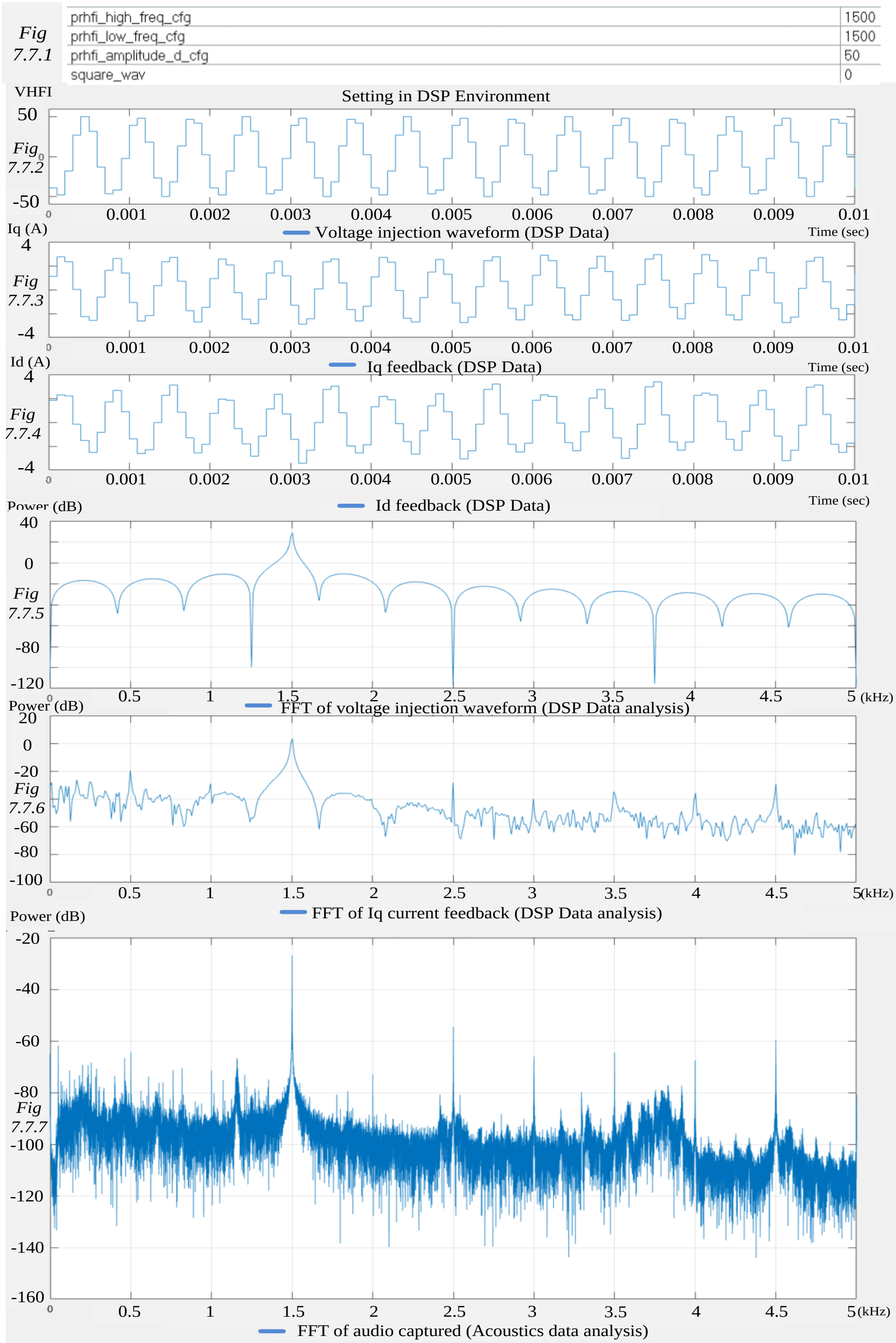


Figure 7.7 DSP logged data and acoustic noise capturing from sinusoidal HFI at 1.5kHz on the HEMAS motor

Constant frequency HFI – 1.7kHz Sinusoidal, DSP and Audio Data

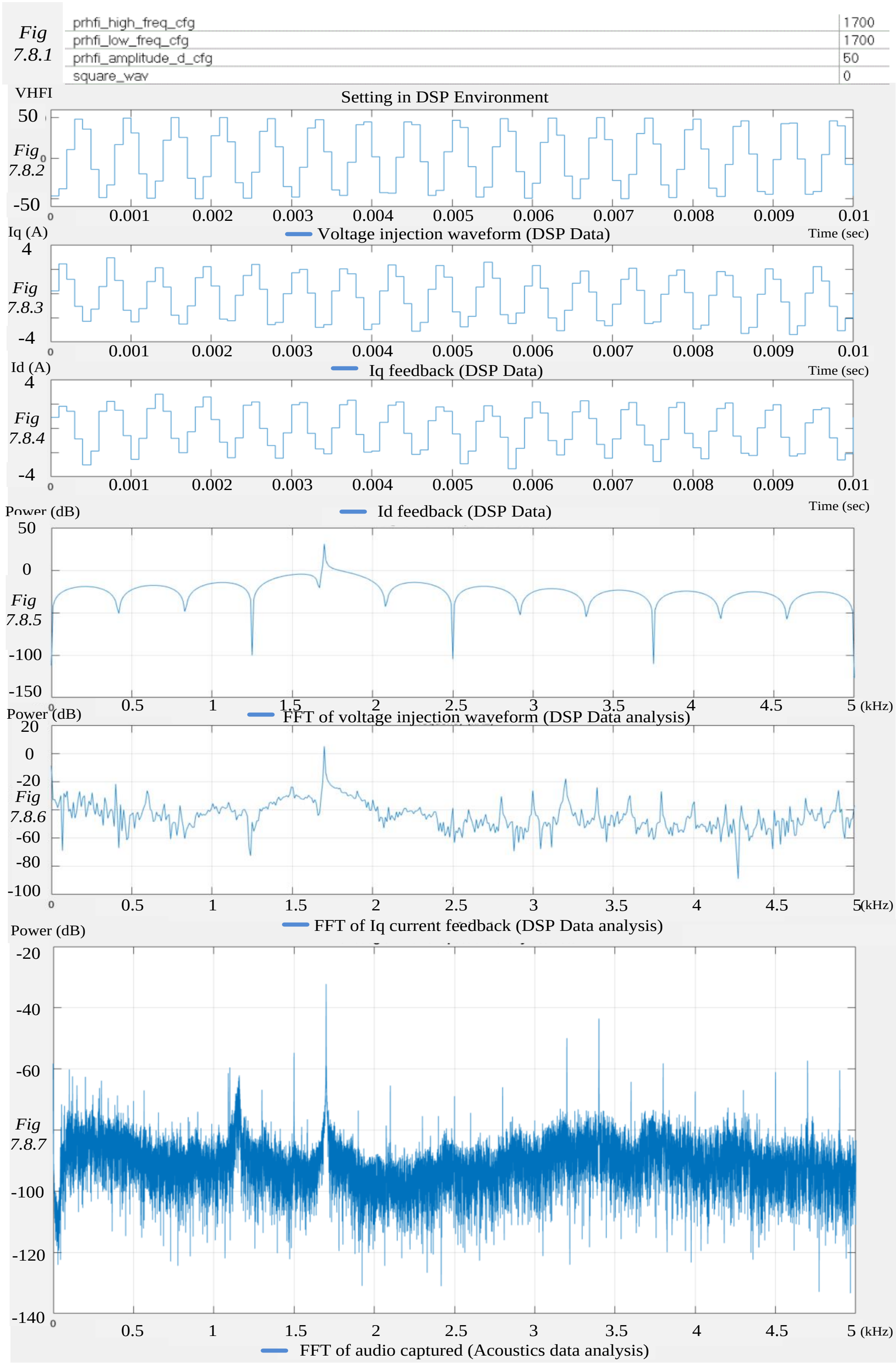


Figure 7.8 DSP logged data and acoustic noise capturing from sinusoidal HFI at 1.7kHz on the HEMAS motor

Constant frequency HFI – 2kHz Sinusoidal, DSP and Audio Data

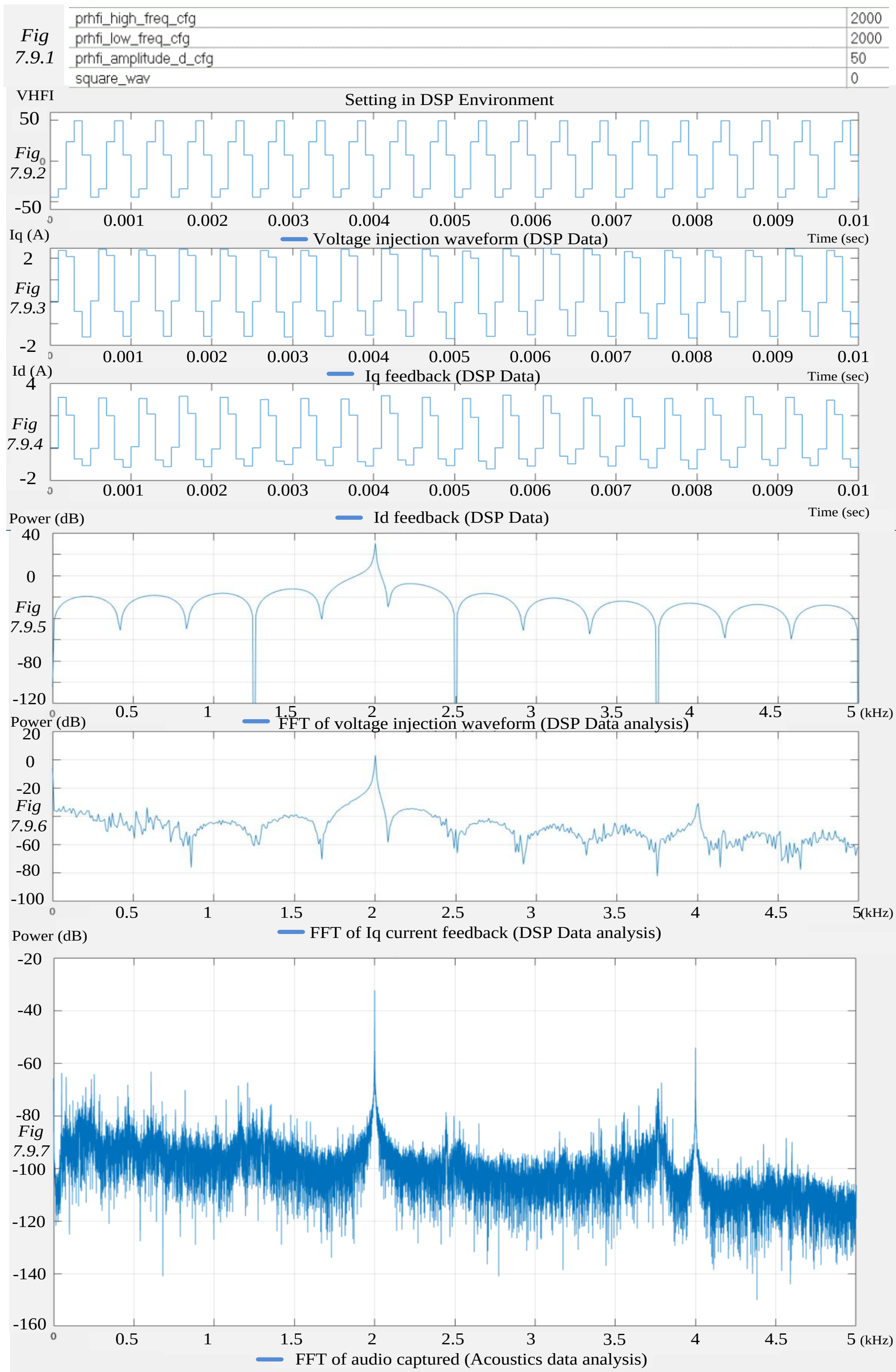


Figure 7.9 DSP logged data and acoustic noise capturing from sinusoidal HFI at 2kHz on the HEMAS motor

Constant frequency HFI – 1kHz Square, DSP and Audio Data

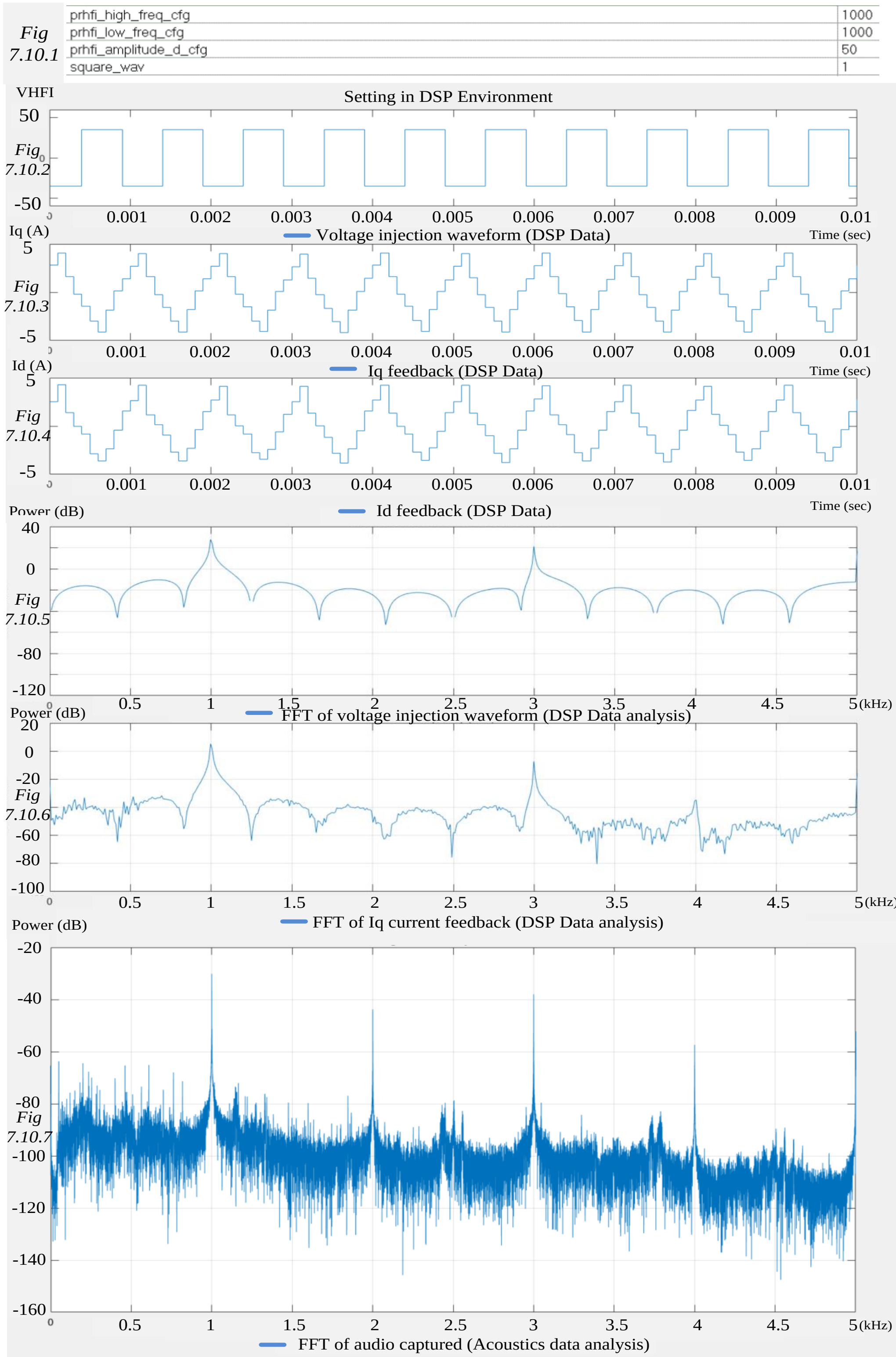


Figure 7.10 DSP logged data and acoustic noise capturing from square HFI at 1kHz on the HEMAS motor

Constant frequency HFI – 1.5kHz Square, DSP and Audio Data

Fig 7.11.1

prhfi_high_freq_cfg	1500
prhfi_low_freq_cfg	1500
prhfi_amplitude_d_cfg	50
square_wav	1

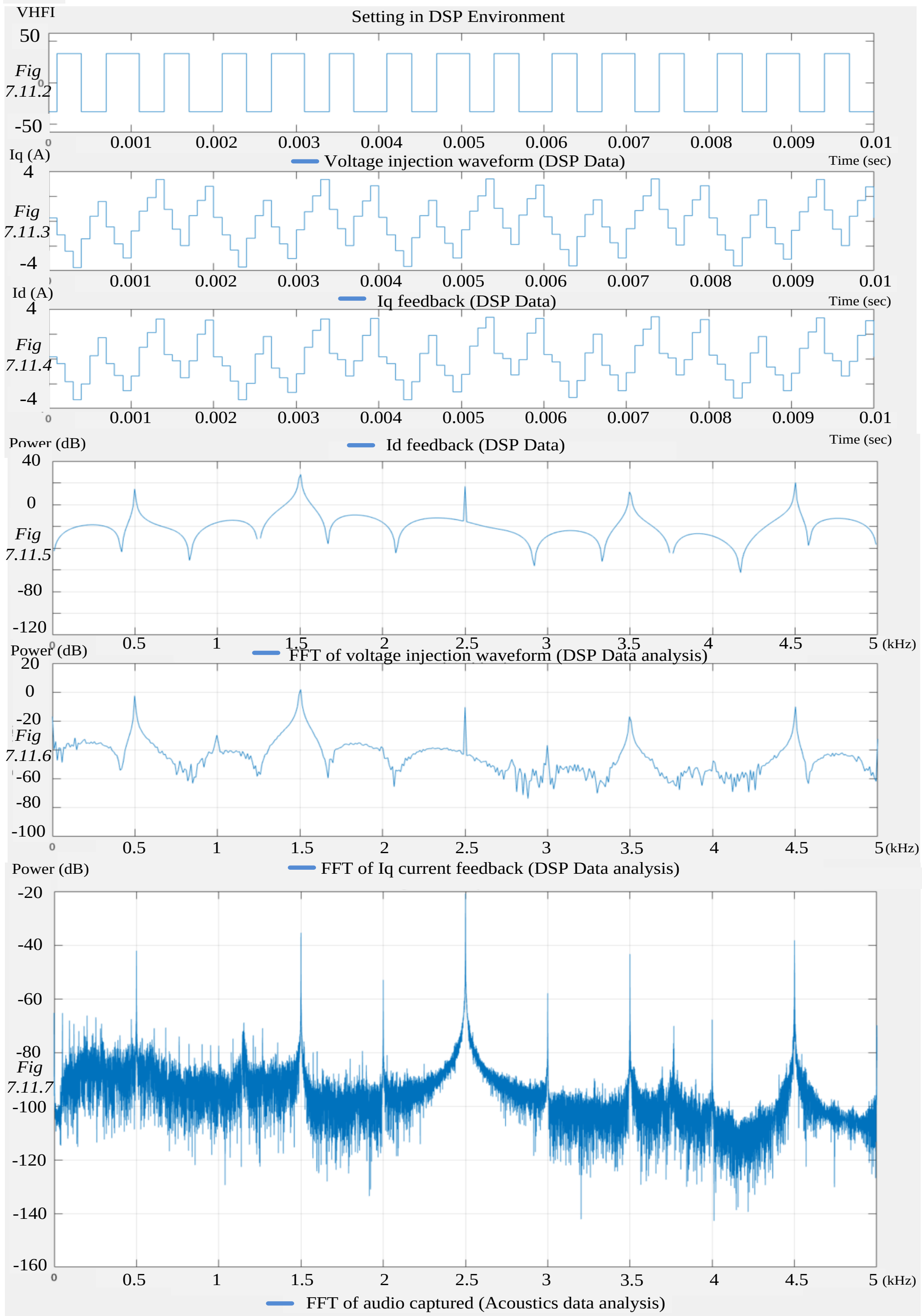


Figure 7.11 DSP logged data and acoustic noise capturing from square HFI at 1.5kHz on the HEMAS motor

While a detailed analysis of the HFI results is presented in section 7.7, one key observation from square wave injection at 1.5kHz needs to be made. A varying DC offset over consecutive injection iterations can be observed in the motor phase currents. This is because the duration of positive and negative square wave voltage demand is not equal and changes over time. The cause of this relates to the PWM frequency being 10kHz i.e. the PWM period being 100 microseconds. The requested 1.5kHz square wave injection has a period of 666 microseconds which cannot fit perfectly using the available PWM period of 100 microseconds. The injection algorithm illustrated in figure 7.1 passes on the remainder injection time to the next injection cycle therefore creating a continuously changing duration between positive and negative voltage demands aiming to reach the unachievable period of 666 microseconds. The advantage of this approach is that over a large number of injection cycles the algorithm maintains the average DC offset close to 0 but the disadvantage is that it introduces current oscillations in consecutive injection cycles that can impact sensorless accuracy. There are a few ways of attempting to correct this issue:

- One way to address the observed momentary offset fluctuation is to not pass the remainder of the time to the next injection cycle and allow the positive square voltage demand to have a fixed and different duration relative to the negative voltage demand. This approach was trialed and testing results are shown in the Figure below. While the oscillation has been reduced the major disadvantage of this method is that there is a constant large DC offset of the introduced current so instead of the current oscillating from $-3A$ to $+3A$ it oscillates from $-4.8A$ to $1.2A$ (note that the injection convention is $-\sin(\omega t)$. This side effect is actually highly undesirable because it creates uneven average positive/negative torque pulsations and can cause unintended spinning of the motor.

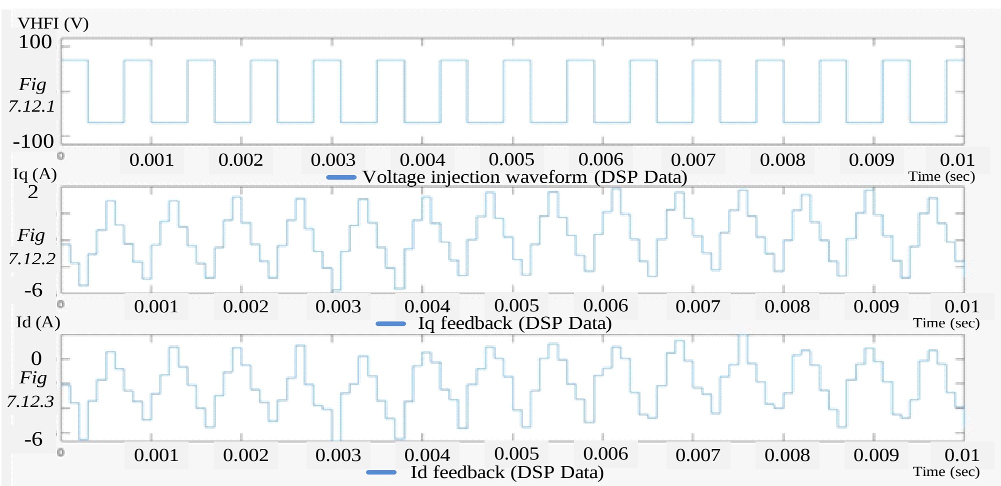


Figure 7.12 DSP logged data from square HFI at 1.5kHz resetting the phase injection angle

- Another option would be for the algorithm to alter the injection frequency versus the desired frequency such that it can fit using the available 100 microseconds PWM period and the positive injection time to be always equal to the negative injection time. For example if the user requested injection of 1.5kHz i.e. period of 666 microseconds it would inject frequency of 1.25kHz instead that has a period of 800 microseconds and formed by an integer multiple of 100 microseconds steps for the positive and negative voltage demand respectively (400 microseconds/400 microseconds). This would resolve the consecutive iteration DC offset and the average DC offset, however it would alter the injection frequency substantially and reduce the number of available injection frequencies.

- A better compromise attempting to address the above issue is to allow the positive and negative injection cycles to be of different duration but ensure that the positive injection cycle has different injection voltage such that that it compensates the difference in its duration versus the negative cycle. This method minimizes the altering of the injection frequency and does not create a short term or long term DC offset in the introduced HF currents.

The issue experienced above and the proposed solution is shown graphically in Figures 7.13.

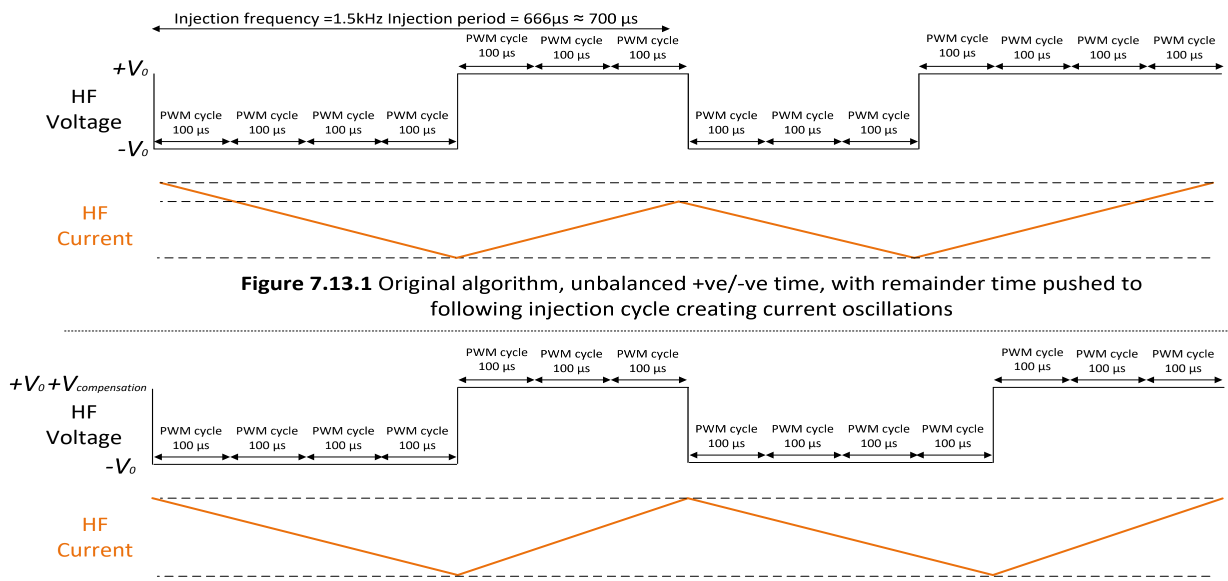


Figure 7.13.1 Original algorithm, unbalanced +ve/-ve time, with remainder time pushed to following injection cycle creating current oscillations

Figure 7.13.2 unbalanced +ve/-ve HF amplitude compensates unbalanced +ve/-ve time

Figure 7.13 PWM side effects of different injection duration for positive and negative cycle

The code was therefore changed for square wave injection from:

```
PRHFI_injection_d = 0.7 * prhfi_amplitude_scaled;  // square +ve voltage
```

To:

```
if (((int)((500000/prhfi_freq)) % 100) == 0)
{    // injection half period multiple of PWM period,no need for compensation
   sq_pos_scaled = 0.7 * prhfi_amplitude_scaled;
}
else // apply amplitude compensation to positive square
{
   sq_pos_scaled = 0.7 * prhfi_amplitude_scaled *
                      ((1/prhfi_freq)/2)/(((1/prhfi_freq)/2)-0.0001);
}
PRHFI_injection_d = sq_pos_scaled;
```

The square wave injection at 1.5kHz testing was subsequently repeated and shown below.

Constant frequency HFI – 1.5kHz Square, DSP and Audio Data (repeated)

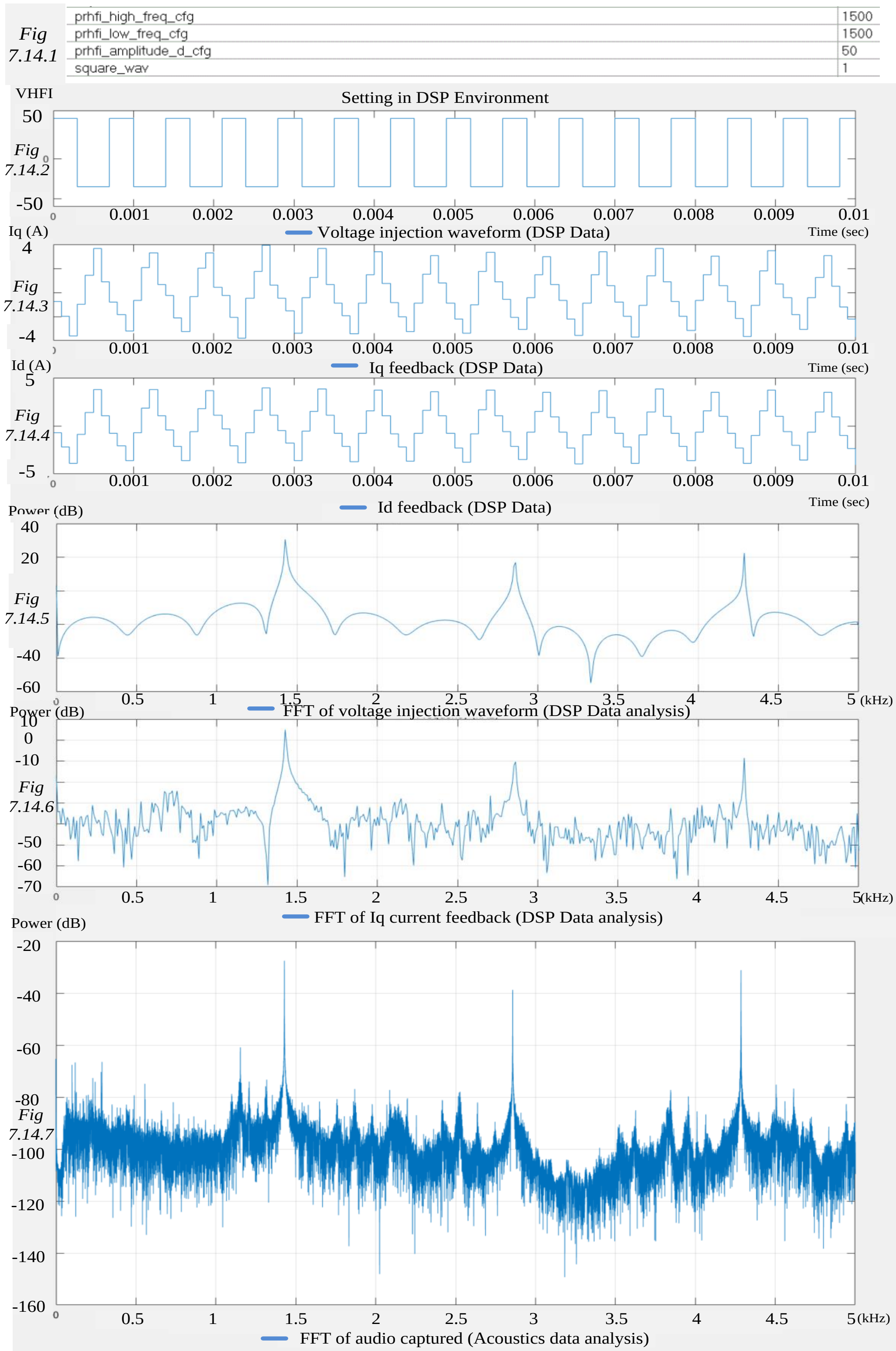


Figure 7.14 DSP logged data and acoustic noise capturing from square HFI at 1.5kHz on the HEMAS motor repeat

Constant frequency HFI – 2.0kHz Square, DSP and Audio Data

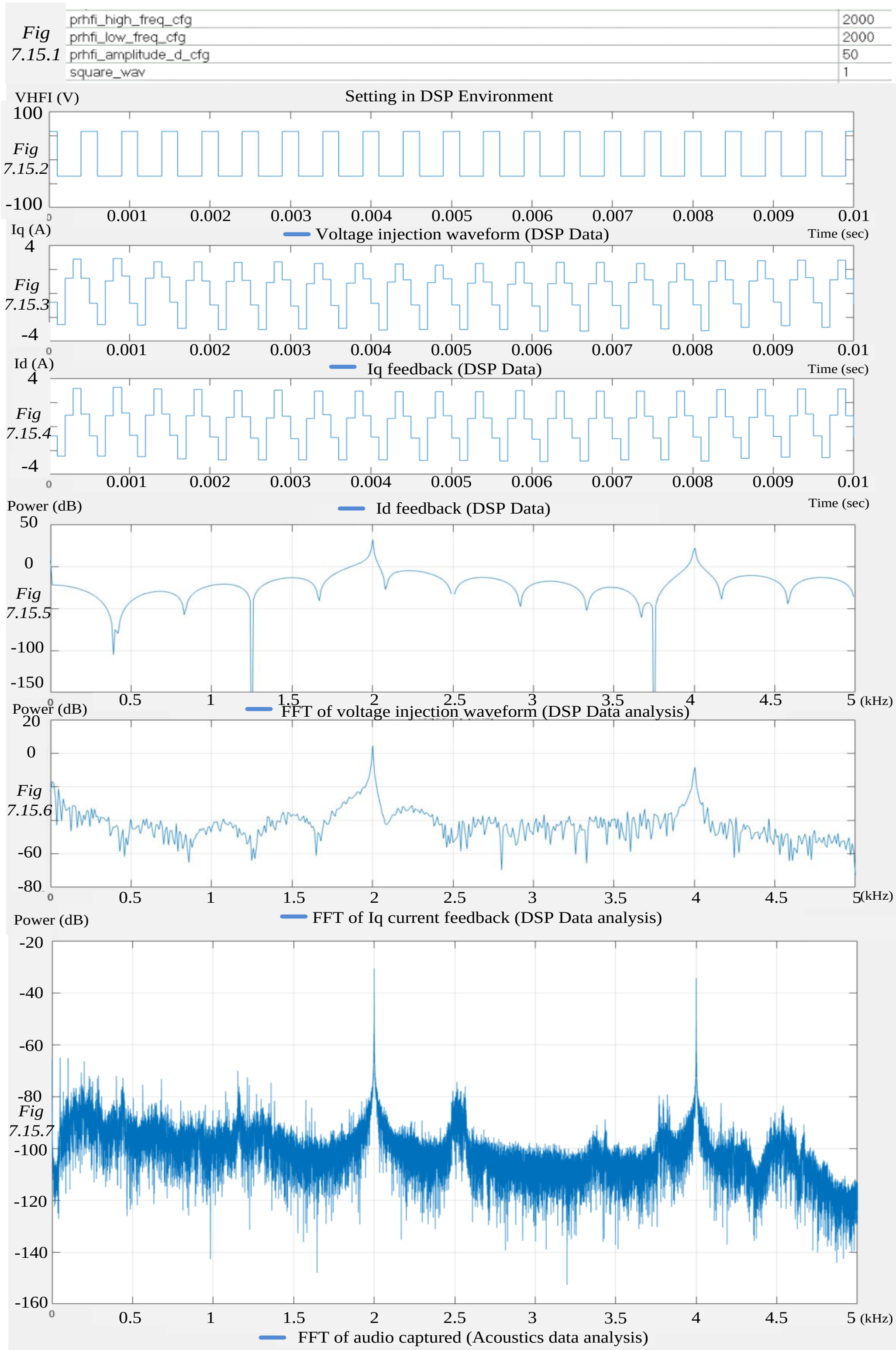


Figure 7.15 DSP logged data and acoustic noise capturing from square HFI at 2.0kHz on the HEMAS motor

Constant frequency HFI – 2.5kHz Square, DSP and Audio Data

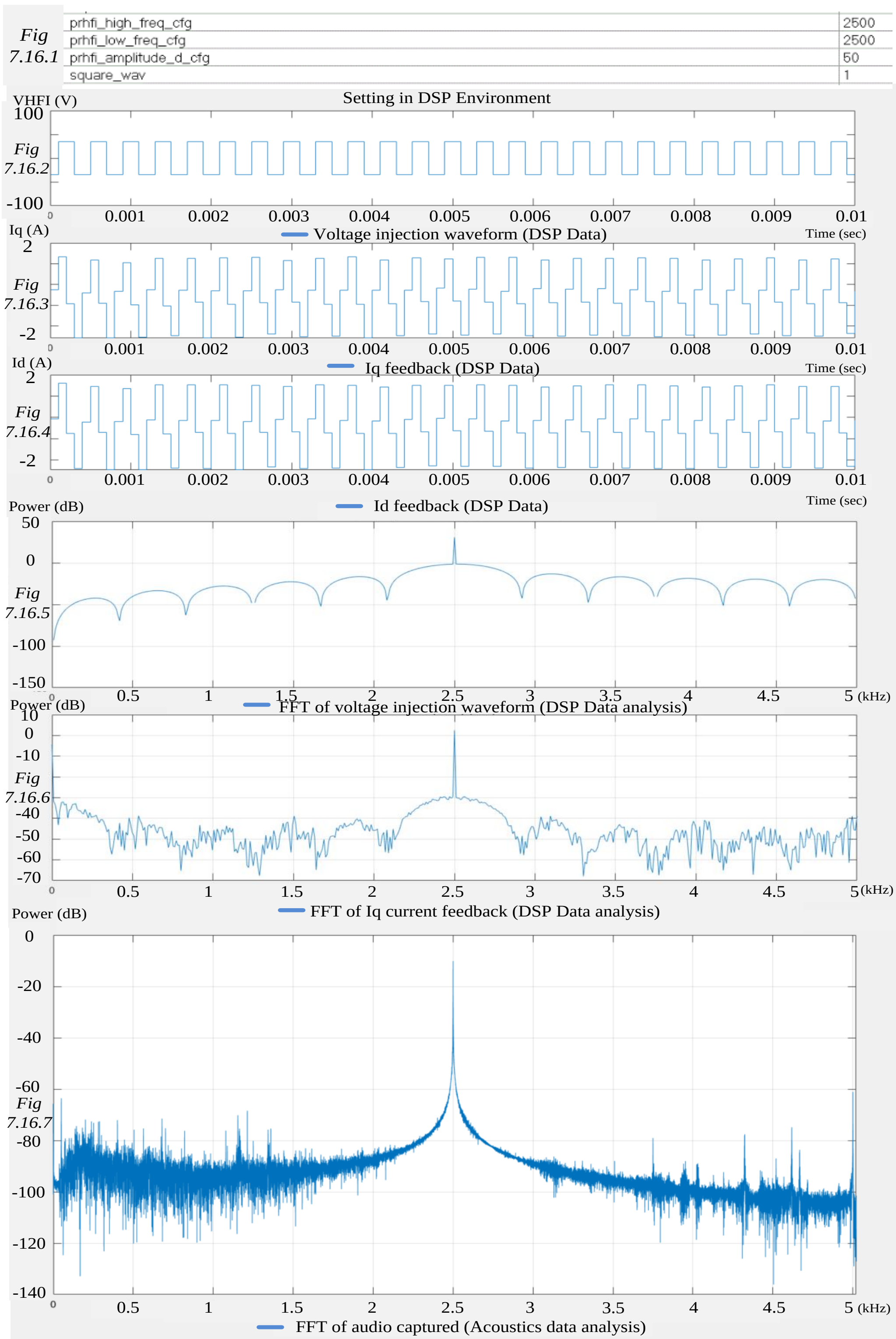


Figure 7.16 DSP logged data and acoustic noise capturing from square HFI at 2.5kHz on the HEMAS motor

7.6.2 DSP data and Acoustics from PRHFI

Aiming to evaluate the proposed PRHFI algorithm, pseudo random injection option was enabled and test results obtained are presented in this section.

PRHFI – 1.9kHz-2.2kHz, DSP and Audio Data

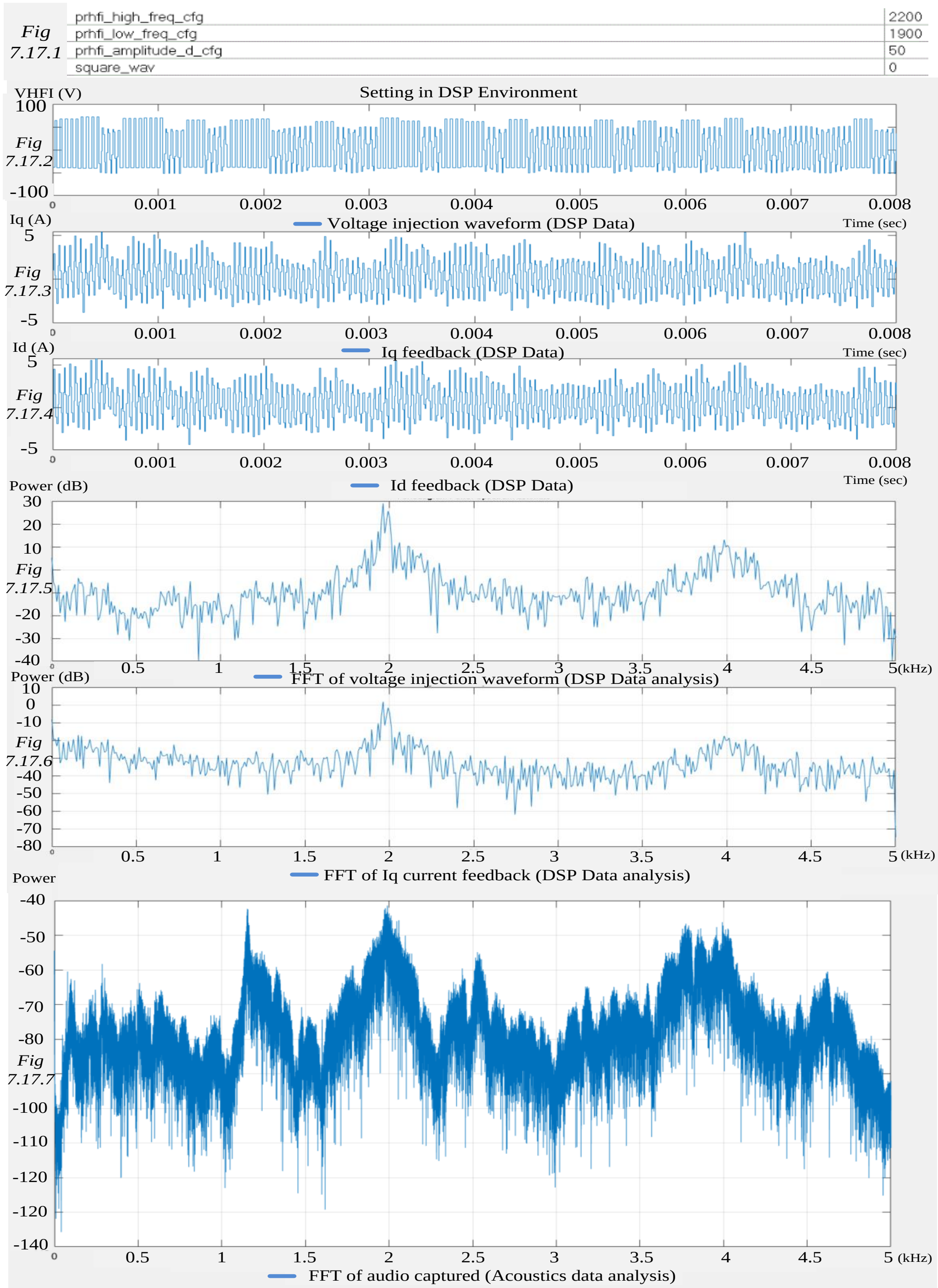


Figure 7.17 DSP logged data and acoustic noise capturing from PRHFI 1.9kHz-2.2kHz on the HEMAS motor

Figure 7.17 illustrates the test data produced by the proposed algorithm when the user selects the injection frequency to be between 1.9kHz and 2.2kHz. A natural question that can be posed by the reader is why this specific frequency range was selected. The motivation for this selection is partly technical related to motor drive electronics and partly due to what was introduced in chapter 3 as "The psychological perception of noise". Maintaining the injection frequency above 1kHz is essential in terms of motor control to maintain current control bandwidth high. Selecting a specific spectrum of audio frequencies above 1kHz was a less obvious and to some extent subjective process. Research [65] and [83] agree that a spectrum of audio frequencies is more pleasant to a human ear than pure tones. Research [84] claims that if audio is purely tonal, the frequencies that appear most irritant to the ear are above 2kHz and specifically 2-5kHz. However although research exists on irritant frequency tones, there is not conclusive research on pleasant Wideband or Narrowband frequency spectra. The author therefore evaluated the sound produced by a number of injection frequency ranges and concluded to the selection of 1.9kHz-2.2kHz, 1.7kHz-2.2kHz, 1.5kHz-2.2kHz based on the appeal to the human ear of the author.

Similarly to constant frequency High Frequency Injection, figure 7.17 illustrates that the spectrum of frequencies of the generated current and acoustic noise align with the user selected frequency range of the algorithm i.e. 1.9kHz to 2.2kHz. A secondary observation is that there is some fluctuation in the generated currents as the algorithm switches between frequencies and shapes. Although the acoustic noise range is not impacted by this limitation, Chapter 9 includes in Future Work improvement of the algorithm maintaining HF currents more stable.

The key conclusion from the PRHFI data is that the algorithm is able to create the spectrum of audio frequencies that were selected by the user i.e.1.9kHz to 2.2kHz as shown in Figure 7.18.

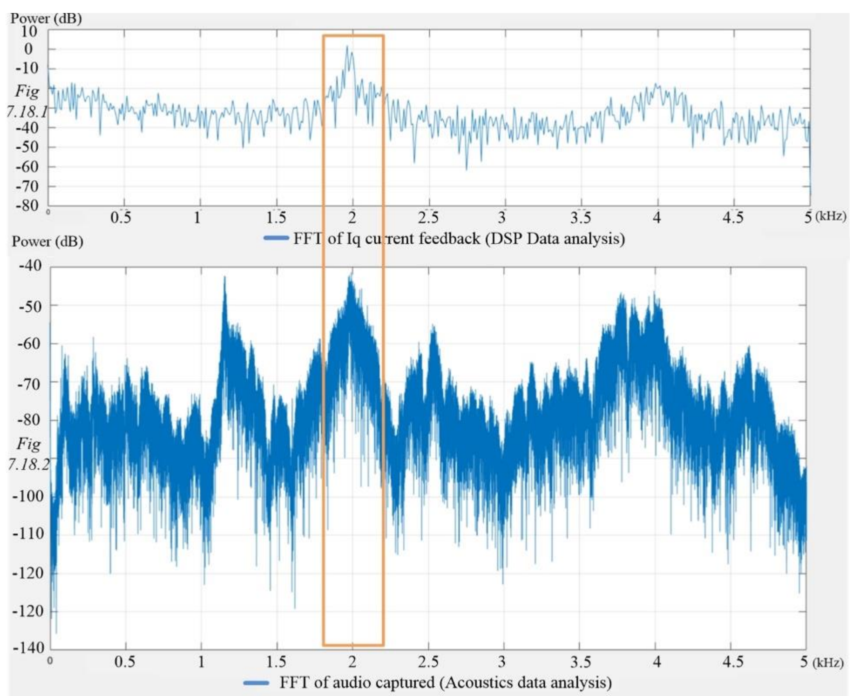


Figure 7.18 Analysis of data from PRHFI at 1.9kHz-2.2kHz on the HEMAS motor

PRHFI – 1.7kHz-2.2kHz, DSP and Audio Data

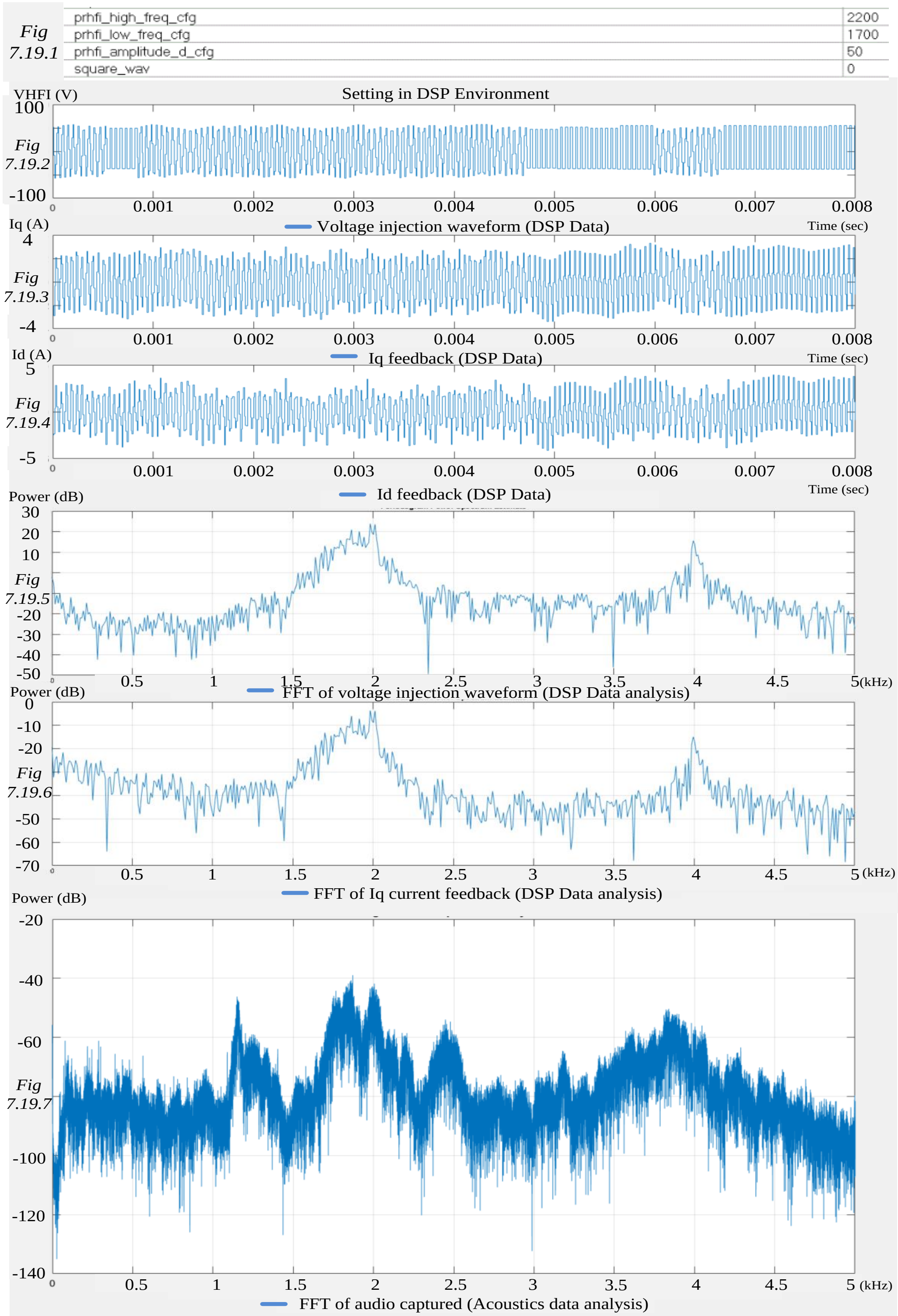


Figure 7.19 DSP logged data and acoustic noise capturing from PRHFI 1.7kHz-2.2kHz on the HEMAS motor
Note that analysis of the above and subsequent test results can be found in section 7.7 of this thesis.

PRHFI – 1.5kHz-2.2kHz, DSP and Audio Data

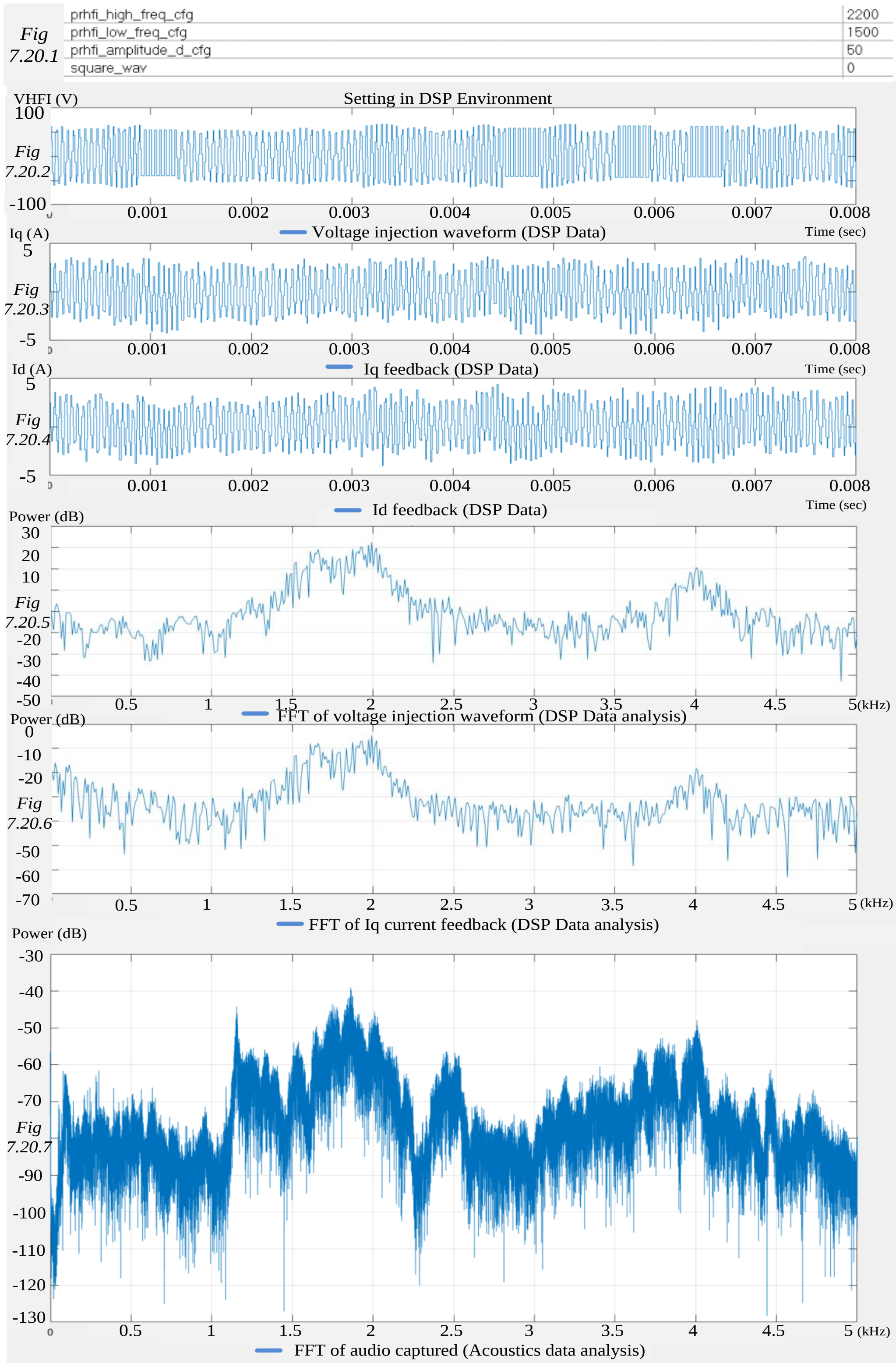


Figure 7.20 DSP logged data and acoustic noise capturing from PRHFI 1.5kHz-2.2kHz on the HEMAS motor

Claiming for an algorithm to be capable of improving acoustics would be incomplete if the reader was not provided the opportunity to evaluate this improvement using his/her hearing sense. The acoustic noise from each of the injection frequencies above as captured using a microphone is available online below:

Captured acoustic noise from constant frequency HFI to the HEMAS platform

Audio link 7.3: HFI sinusoidal 1kHz:	https://youtu.be/46tx2-DxwD8
Audio link 7.4: HFI sinusoidal 1.4kHz:	https://youtu.be/h57xReLYomQ
Audio link 7.5: HFI sinusoidal 1.5kHz:	https://youtu.be/4MJPTypFDd0
Audio link 7.6: HFI sinusoidal 1.7kHz:	https://youtu.be/A5vfLuyY6Ns
Audio link 7.7: HFI sinusoidal 2kHz:	https://youtu.be/eN_xloZQHQI
Audio link 7.8: HFI square 1kHz:	https://youtu.be/r_l5PwabKGQ
Audio link 7.9: HFI square 1.5kHz:	https://youtu.be/nvA1mAoGRZQ
Audio link 7.10: HFI square 2kHz:	https://youtu.be/UdCBqQLIrI0
Audio link 7.11: HFI square 2.5kHz:	https://youtu.be/qg3QNZQ-L1c

Captured acoustic noise from Pseudo Random HFI to the HEMAS platform

Audio link 7.12: PRHFI 1.9kHz-2.2kHz:	https://youtu.be/1FTgSl6G4uM
Audio link 7.13: PRHFI 1.7kHz-2.2kHz:	https://youtu.be/H99WrcrY9Yc
Audio link 7.14: PRHFI 1.5kHz-2.2kHz:	https://youtu.be/twiF9Rrbqek

Section 7.7 below attempts to analyse the experimental results presented above and reach key findings.

7.7 Analysis of test results

Figures 7.4-7.18 presented the DSP and acoustic data from constant and Pseudo Random High Frequency Injection covering a range of frequencies and waveform shapes. The first and a rather significant observation is a HFI voltage of frequency n kHz produces current, torque and acoustic noise centered at the same frequency, a correlation that is shown more clearly in the concentrated results Figure 7.21. This observation validates the analysis performed in section 7.2 of this thesis. A second observation is that there are harmonics of lower amplitude in the current, torque and acoustics. These harmonics exist for a number of reasons specifically due to the PWM switching frequency, device non-linearities and the acoustic sampling process. As the PWM switching frequency of 10kHz is close to the HF injection frequency, the voltage steps cannot construct a perfect sinusoid and the resultant currents are non-sinusoidal appearing as harmonics in the currents and acoustics FFT. In a similar fashion the current becomes distorted due to inverter component non linearities and the sampled audio becomes distorted further during the digitization process by the microphone.

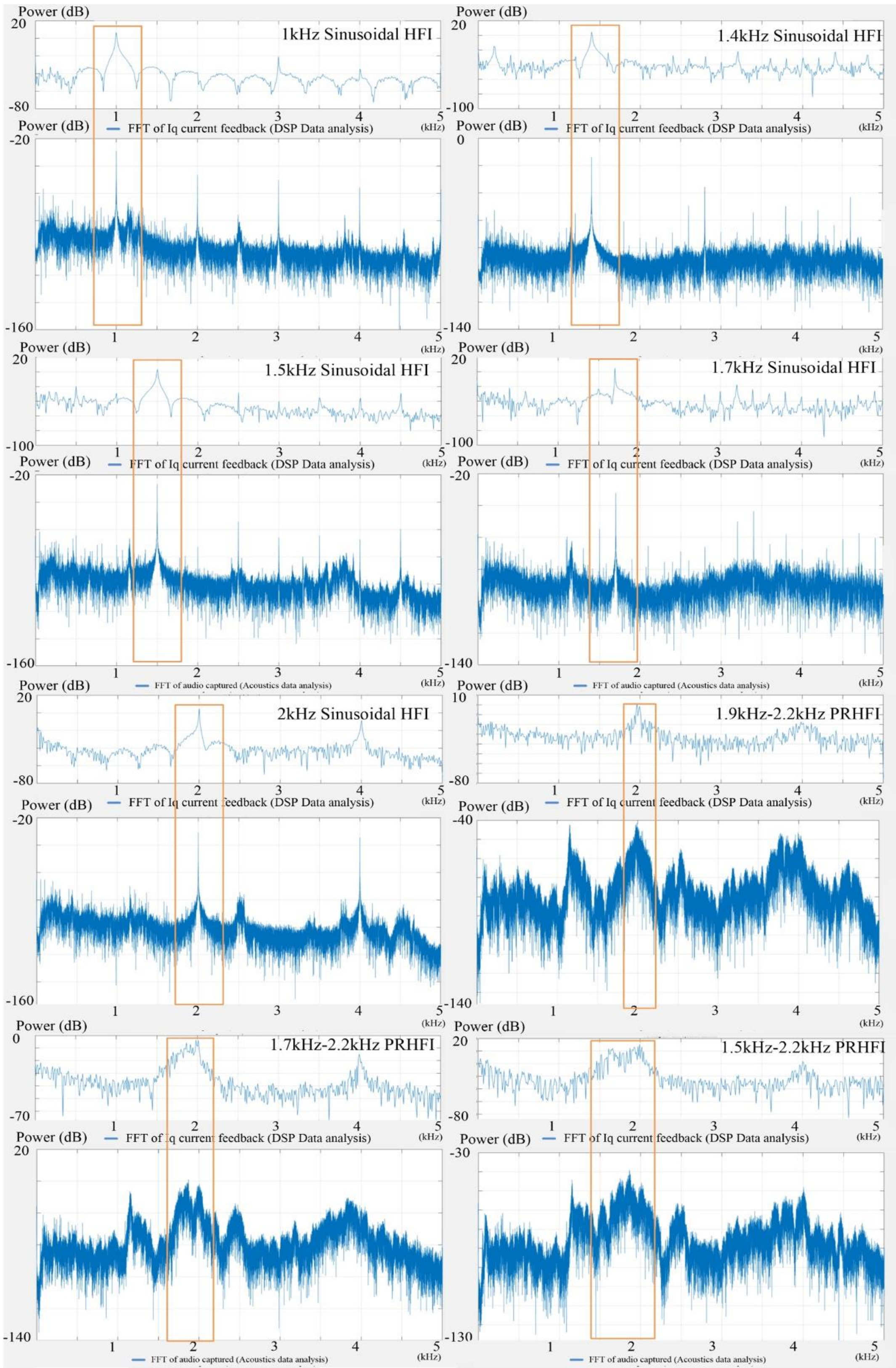


Figure 7.21 Analysis of data and acoustic noise capturing from HFI and PRHFI on the HEMAS motor

Another minor comment on the test data is that the acoustic FFT appears more accurate than the DSP data FFT. This is because the former is sampled at 48kHz with a large number of samples while the latter is sampled at 10kHz with a limited number of samples due to hardware limitation of the DSP at the controller.

The key and most significant question attempted to be answered in this chapter is whether an improvement has been obtained from the proposed PRHFI algorithm. Evaluating and comparing the acoustics between constant frequency HFI and PRHFI as presented in web audio links, one observation is that listening to tonal noise (1kHz, 1.5kHz, 2kHz sinusoidal/square) over prolonged amount of time for example during a long-haul flight can be difficult to bear or based on existing research [85] it can even result into headaches/migraines. Changing the acoustic noise frequency to a wider spectrum and selecting a frequency spectrum that sounds appealing to the human ear for example PRHFI 1.9kHz-2.2kHz can make the generated noise far more pleasant and possible to withstand over long time.

7.8 Conclusions

This chapter presented a variant of the Pseudo Random High Frequency Injection algorithm aiming to improve the acoustics and perception of noise from High Frequency Injection. It provided the details of the proposed PRHFI algorithm, followed by analytical and test results when applied to the HEMAS system. Hardware testing first validated the anticipated relation between injection frequency and acoustics. Subsequent testing illustrated that acoustic improvements can be obtained when the proposed PRHFI algorithm is used instead of a constant frequency HFI method. The proposed method can thus be used to reduce the perception of noise and enable sensorless methods in cases where tonal noise would be prohibitive in terms of acoustic comfort to humans.

Chapter 8 Active Noise Cancellation by means of HFI

8.1 Introduction

Reducing the perception of acoustic noise from HFI was investigated in chapter 7 of this thesis. One interesting scientific question that was posed during this research was what if the noise from HFI was not suppressed but was controlled and enhanced to achieve Active Noise Cancellation (ANC) within a system. The investigation of this idea led into the development of the novel method presented in this chapter that was named Active Noise Cancellation (ANC) by means of High Frequency Injection (HFI). This chapter presents the proposed method along with analytical and experimental test results when applied to the HEMAS platform. Along with the novelty aspect of the method, taking into account the increased levels of electrification in a range of industrial sectors, the proposed concept could have a significant impact in acoustics of systems deploying motor drives.

8.2 Conventional Active Noise Cancellation schemes

Active Noise Cancellation/Control (ANC) is the method of introducing an acoustic wave that is in anti-phase relation to an existing noise source within a system, aiming for the two waves to cancel each other out and result into a quieter acoustic system [86] (See Figure 8.1.1). An ANC system would typically be composed of a set of microphones capturing the existing noise, a digital signal processing system (DSP/microprocessor, FPGA) analysing the captured sound and a speaker producing anti-noise i.e. the wave that is in anti-phase relation to the original acoustic noise [86].

Such ANC systems are encountered in a variety of applications from low-end noise cancellation headphones to state of the art noise cancelling systems in aircraft cabins. An example of the latter application is a family of systems developed by Ultra for a number of aircrafts (Bombardier, Lockheed Martin, Beechcraft King Air and Saab) [87]. However, the above mentioned aircraft ANC systems, require the installation of microphones and speakers within the cabin. The proposed HFI ANC method does not necessitate installation of speakers or specialized actuators and attempts to cancel noise at its source rather than at a distance.

8.3 The proposed HFI ANC method

8.3.1 Description of the method

The HFI ANC method proposed in this thesis involves superimposing a High Frequency voltage and controlling its frequency and phase so as to be in anti-phase relation and cancel an existing noise (See Figure 8.1.2).

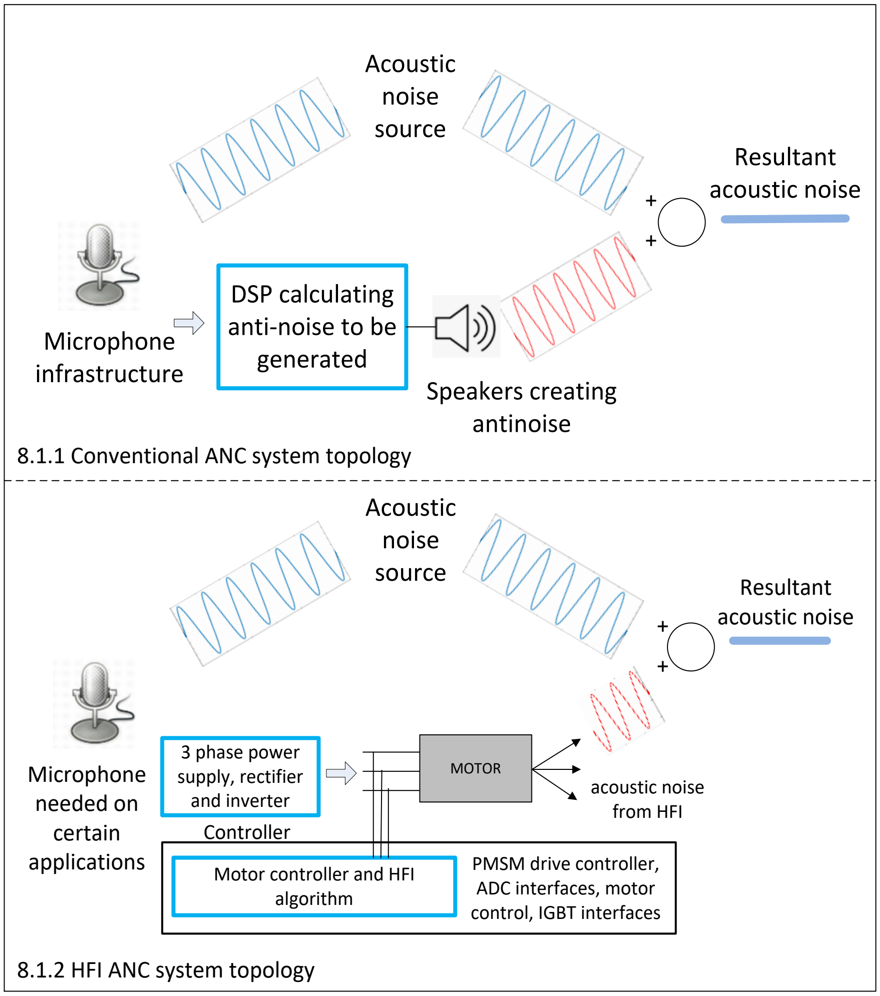


Figure 8.1 High Frequency Injection for a 3 phase motor

There are cases where the residing acoustic noise to be cancelled has a predictable pattern for example an electric compressor that experiences load and acoustic pulses at a predefined mechanical angle. In such cases the HFI ANC method can be applied without the need of a microphone using a feed forward open loop method. In cases that the acoustic noise is more random in nature the HFI ANC method is best applied in conjunction with a microphone. The motor controller in that case along with managing the spinning of the motor it would also process the captured audio and inject anti-phase sound waves using HFI.

8.3.2 Advantages of the proposed HFI ANC method

The advantages of the proposed method versus conventional ANC systems can be summarized to the below:

• ANC capability without the need of speaker installation: the proposed ANC method re-uses the same hardware infrastructure used to spin a motor and adds a noise cancellation capability. The method can therefore reduce cost, save space and reach difficult to approach places such as the propeller of an aircraft or of a submarine.

• ANC at the source of noise: being able to create anti-noise using a motor drive can cancel acoustic noise at its source in case of the propeller noise of an electric/hybrid aircraft rather than at a distance. This can be beneficial as the existing commercial cabin noise cancellation systems attempt to address the noise within the cabin for the passengers but do not try to address the noise emitted to the environment an issue that is more evident near airports.

8.3.3 Example applications

An example list of applications where the HFI ANC method can be applied to is presented below and shown diagrammatically in Figure 8.2:

- Propeller Noise (Fig 8.2.1): HFI ANC is proposed to be used towards the cancellation of propeller noise for an aircraft or submarine. Propeller noise is of special interest as it is an upcoming hurdle for emerging electric/hybrid aircraft propulsion and a known problem in submarine propulsion. Research effort [88] and commercial products by Ultra [87] are two existing examples of aircraft propeller ANC while [89] is an example of submarine propeller ANC. However, the above methods involve the use of specialized speakers and specialized vibratory devices to create anti-noise rather than re-using a motor drive also used for spinning as is the proposed approach of HFI. As introduced in section 3.4 the noise generated from a propeller has both tonal and wideband noise frequency elements. The tonal elements are due to two phenomena termed thickness and loading noise. As tonal noise is a significant part of propeller noise [44] the experiments in this chapter attempt to cover tonal Active Noise Cancellation.
- Noise from a pair of sensorless motor drives (Fig 8.2.2): the HFI ANC method may have in some applications dual function, firstly towards conventional sensorless position calculation and secondly for Active Noise Cancellation purposes. Under this example HFI is used by two sensorless motor drives and their injection voltage is of the same frequency but of anti-phase relation resulting into the acoustic noise of each motor drive naturally cancelling each other without the need of a microphone.
- Noise from connected load or external subsystem (Fig 8.2.3): under this example the acoustic noise to be cancelled by the method originates from the connected load, the gearbox or an external subsystem. This noise once again can be either random or predictable in nature (compressor application load pulse as a function of the mechanical angle).

This chapter intends to illustrate the core function of the HFI ANC method rather than presenting a fully developed commercial system. It therefore focuses on demonstrating its effectiveness against tonal noise that represents certain applications and in the future the scheme could be expanded and scaled up to a more complete and dynamic system.

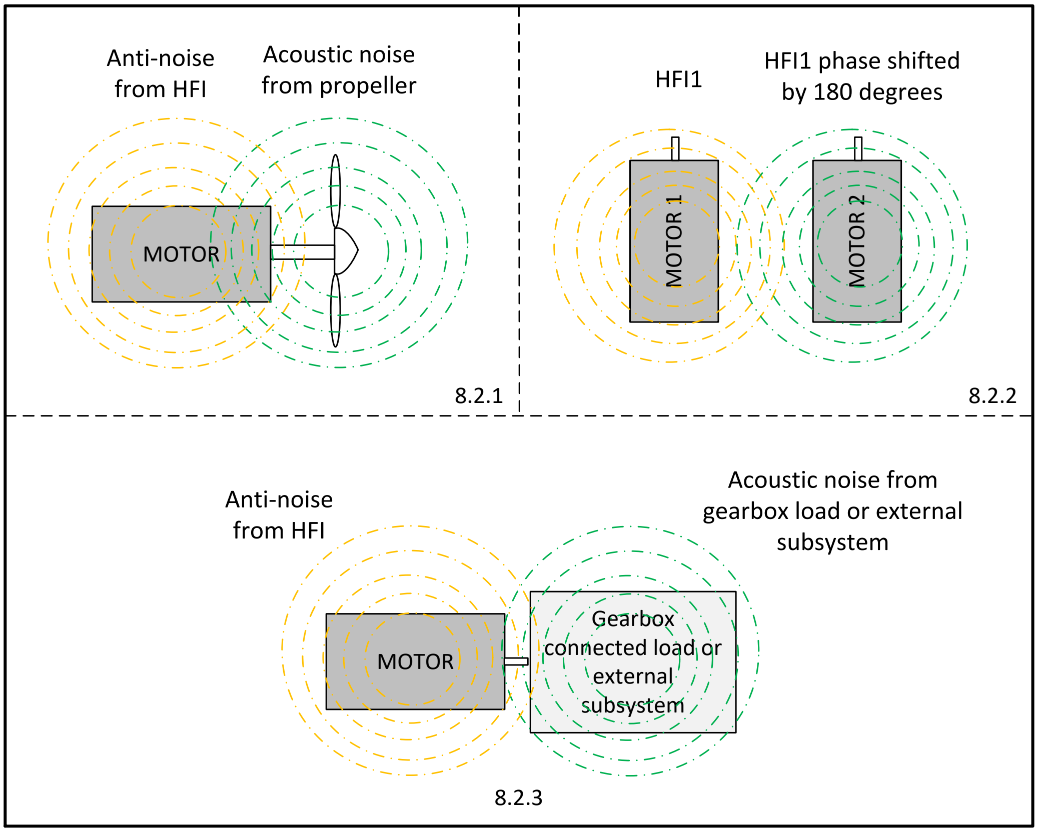


Figure 8.2 Example applications of HFI ANC method

8.4 Test plan for the HFI ANC method

Taking under consideration from Chapter 3 that a significant portion from propeller noise is tonal, the focus of ANC testing in this chapter is on constant frequency noise although it can be expanded in the future to wideband noise cancellation as well. The testing proposed for the method can be summarized to the two experiments below:

Experiment1 HFI ANC tonal testing: to illustrate the effectiveness of the method this experiment is performed by contaminating the system with tonal noise from a speaker and allowing HFI to perform Active Noise Cancellation monitoring the resultant acoustic noise using a microphone. This experiment is to be repeated for a number of frequencies and its set up is shown in Figure 8.3.

Experiment2 HFI acoustic versatility evaluation: while experiment1 is intended to show the effectiveness of the proposed HFI ANC method, experiment2 attempts to convince that a motor drive can be a truly versatile acoustic generation tool. The HEMAS code is thus updated in this experiment to produce musical notes of a classical song so that the motor drive acts as a musical instrument.

The acoustic capturing of both experiments are analysed and provided to the reader in web audio links assisting him/her to reach a conclusions on the effectiveness of the proposed method.

8.5 Analysis and Hardware testing

8.5.1 Experiment 1: HFI ANC using sliding frequencies

This test is performed by contaminating the system with a tonal noise from a speaker and allowing HFI to perform Active Noise Cancellation while monitoring the resultant acoustic noise using a microphone. The set up and topology for this experiment is shown in Figure 8.3 where a laptop and a speaker is brought to close proximity of the HEMAS test rig and play back tonal noise at frequency $f_{speaker}$. HFI is then applied to the HEMAS test rig centred at frequency f_{HFI} initiating the ANC process to take place. The desired relation of the two acoustic waveforms (speaker/HEMAS HFI) for the cancellation to take place would be $f_{speaker} = f_{HFI}$ and the two waveforms to be in anti-phase (180^0 phase offset to each other). However as it would be difficult to clearly prove that this phase relationship holds throughout the test, a more intuitive approach is followed. The two frequencies $f_{speaker}, f_{HFI}$ are slightly different to each other by such a degree that one waveform naturally slides versus the other within a predefined time duration. Specifically if $f_{speaker} =$ 1499.9Hz and f_{HFI}=1500.0Hz then the two waveforms are in anti-phase relationship once every 10 seconds and for the rest of the time they slowly slide versus each other. Similarly if $f_{speaker}$=1499.8Hz and f_{HFI}=1500.0Hz this sliding behaviour takes place once every 5 seconds. A number of $f_{speaker}/f_{HFI}$ frequency pairs has been selected to be tested on the HEMAS platform aiming to illustrate this cancellation process over a range of frequencies. A Matlab simulation model illustrating this acoustic sliding behavior was also developed and can be found in appendix B.1.10 of this thesis. This Matlab simulation is presented in section 8.5.1.1 and aims to provide a golden reference of the anticipated test results.

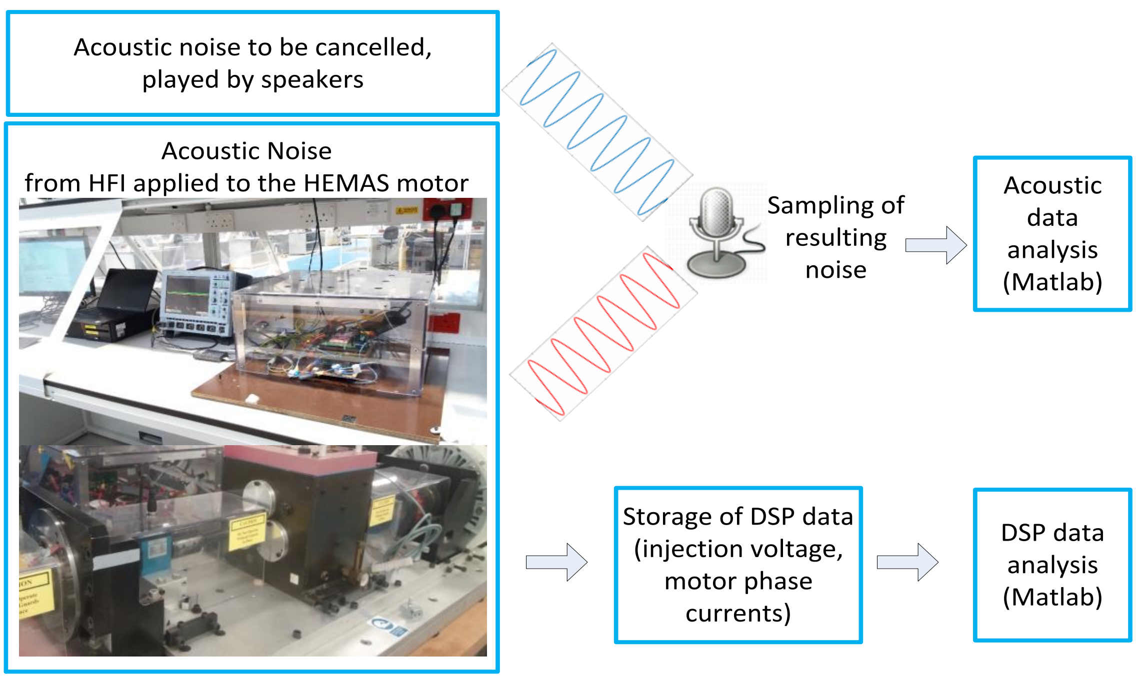


Figure 8.3 HEMAS HFI ANC test set up

The list of frequency pairs ($f_{speaker}$, f_{HFI}) to be tested in this test is presented in Table 8.1 below illustrating the pairs of acoustic frequencies (speaker/HFI) along with the anticipated Active Noise Cancellation period i.e. how often peaks/troughs are to be observed.

$f_{speaker}\ (Hz)$	$f_{HFI}\ (Hz)$	$ANC\ Period$
1399.9	1400	10 sec
1399.8	1400	5 sec
1499.9	1500	10 sec
1499.8	1500	5 sec
1699.9	1700	10 sec
1699.8	1700	5 sec
1999.9	2000	10 sec
1999.8	2000	5 sec

Table 8.1. HFI ANC sliding frequency test table

8.5.1.1 Analysis of anticipated sliding acoustic process

Aiming to illustrate analytically the expected ANC period for each of the acoustic frequency pairs listed in Table 8.1, a Matlab model was written that can be found in Appendix B.1.10 of this thesis. This Matlab model was therefore used to produce the analytical results presented in the figures below.

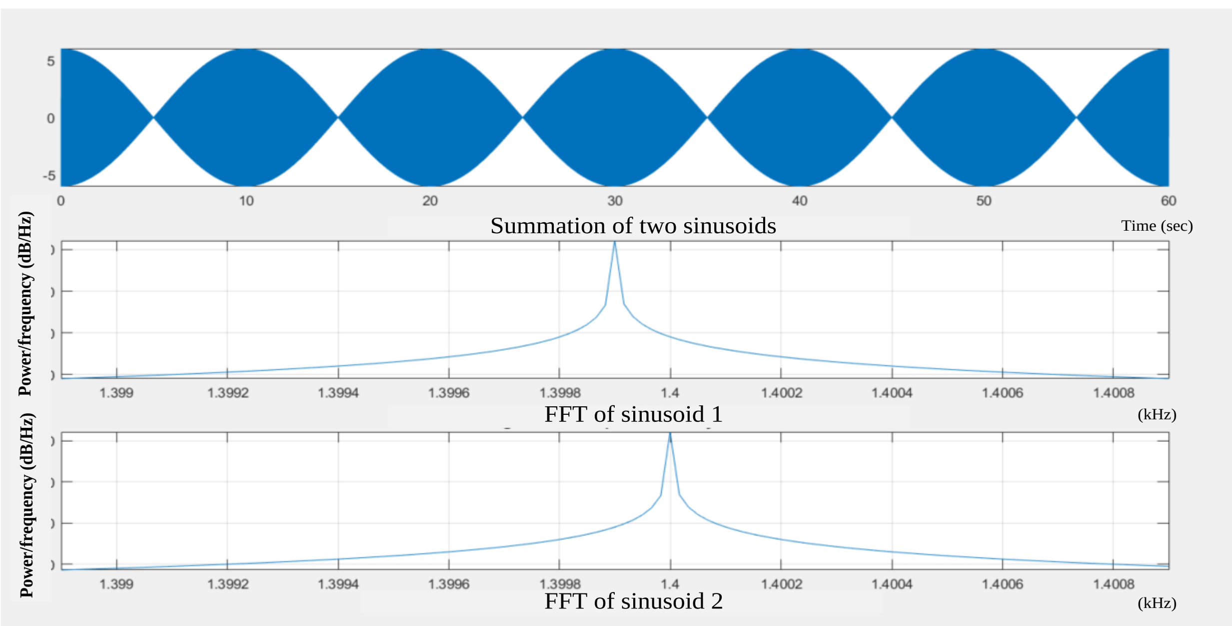


Figure 8.4 Summation simulation of 1399.9Hz and 1400Hz sinusoids

Matlab analysis as illustrated in Figure 8.4 shows that the acoustic summation of 1399.9Hz and 1400Hz introduces a modulated sinusoid with period of 10 seconds.

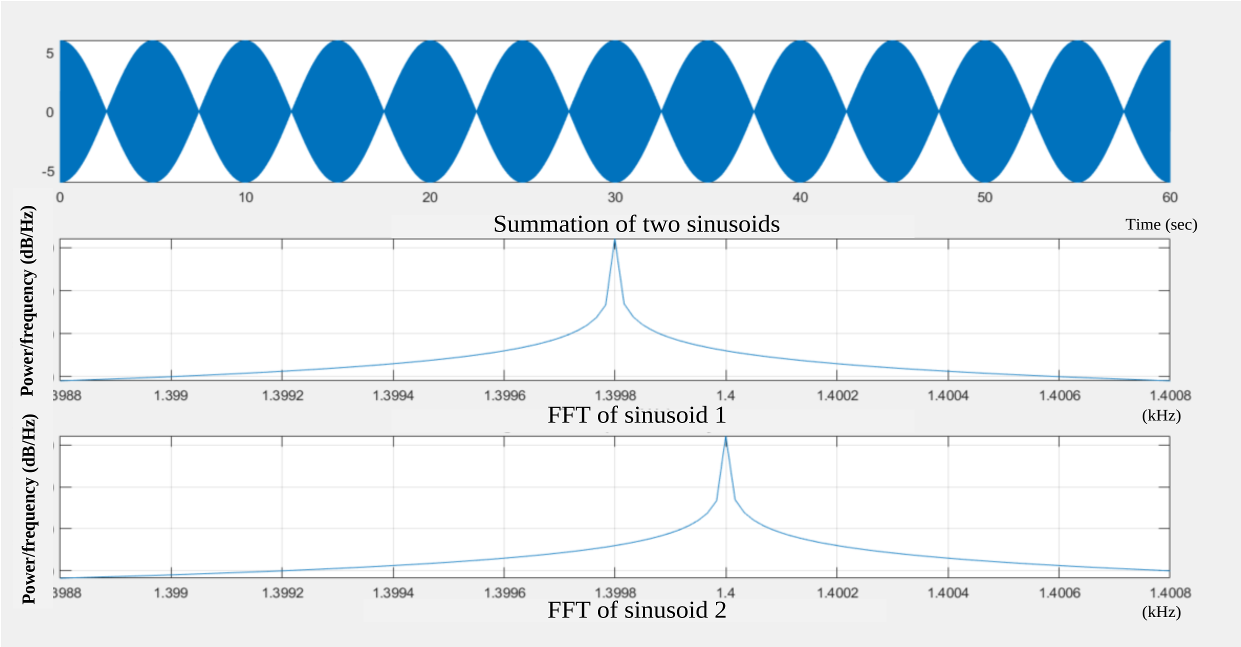


Figure 8.5 Summation simulation of 1399.8Hz and 1400Hz sinusoids

Matlab analysis as illustrated in Figure 8.5 shows that the acoustic summation of 1399.8Hz and 1400Hz introduces a modulated sinusoid with period of 5 seconds.

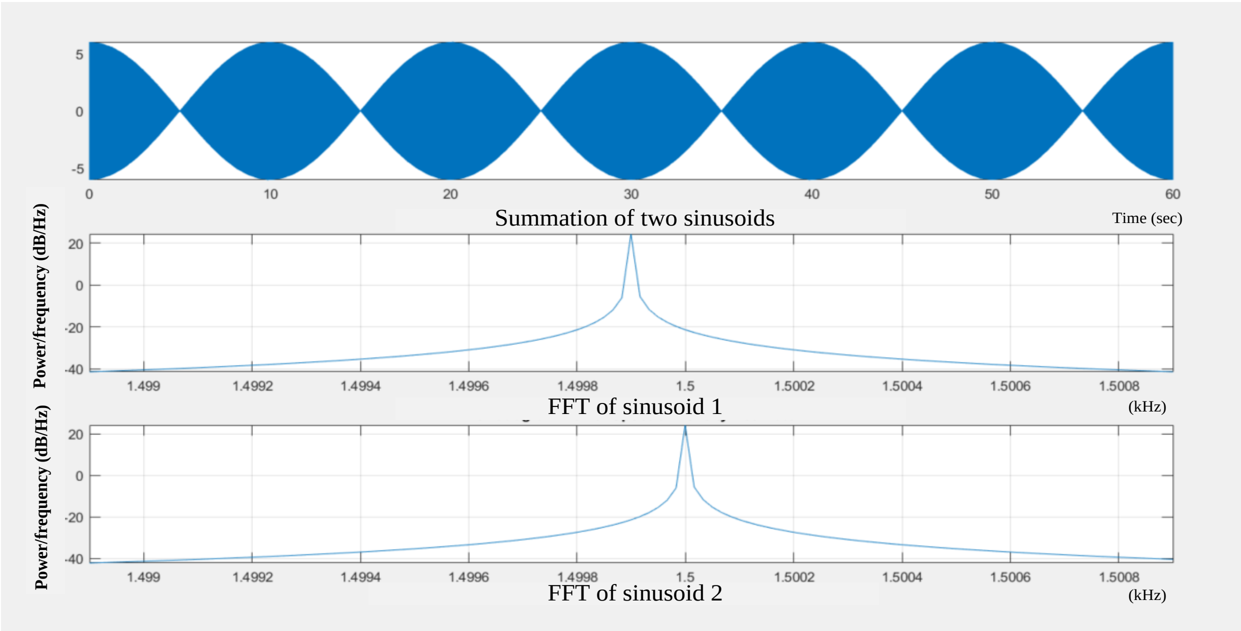


Figure 8.6 Summation simulation of 1499.9Hz and 1500Hz sinusoids

Matlab analysis as illustrated in Figure 8.6 shows that the acoustic summation of 1499.9Hz and 1500Hz introduces a modulated sinusoid with period of 10 seconds.

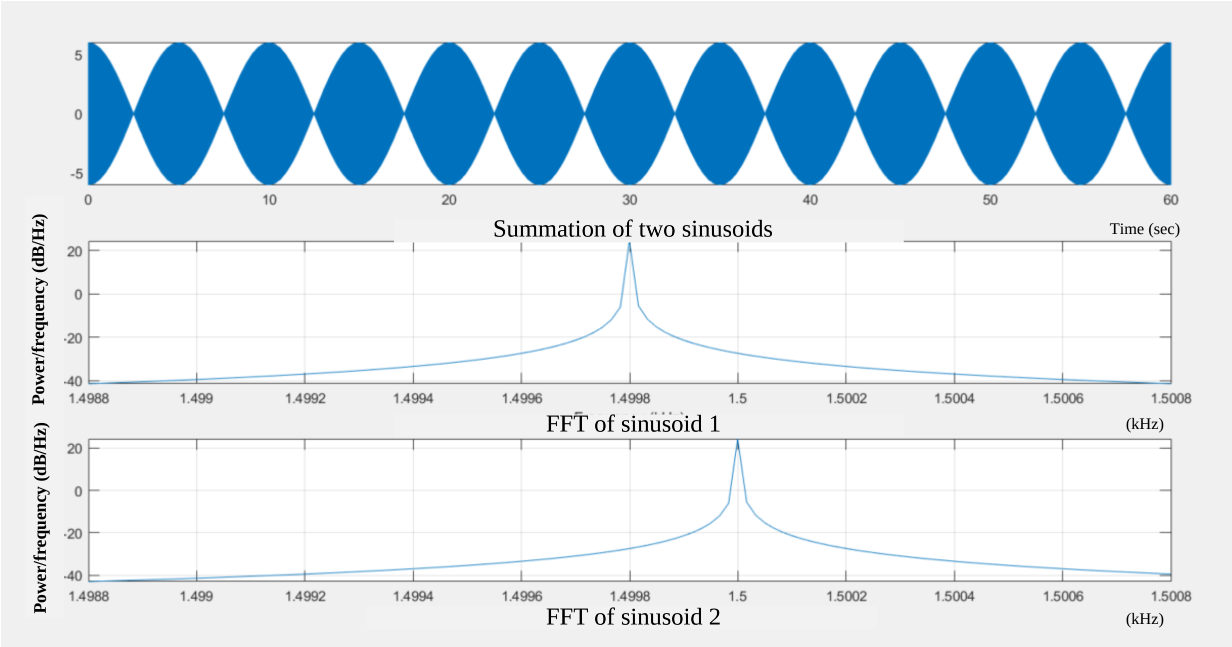


Figure 8.7 Summation simulation of 1499.8Hz and 1500Hz sinusoids

Matlab analysis as illustrated in Figure 8.7 shows that the acoustic summation of 1499.8Hz and 1500Hz introduces a modulated sinusoid with period of 5 seconds.

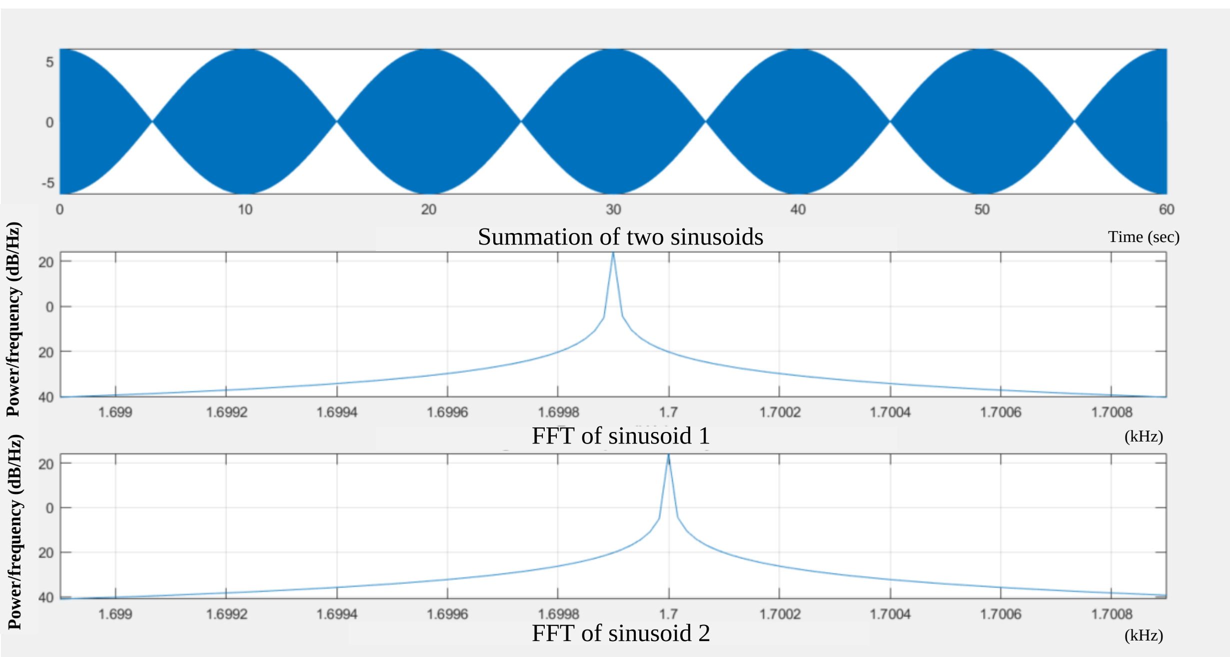


Figure 8.8 Summation simulation of 1699.9Hz and 1700Hz sinusoids

Matlab analysis as illustrated in Figure 8.8 shows that the acoustic summation of 1699.9Hz and 1700Hz introduces a modulated sinusoid with period of 10 seconds.

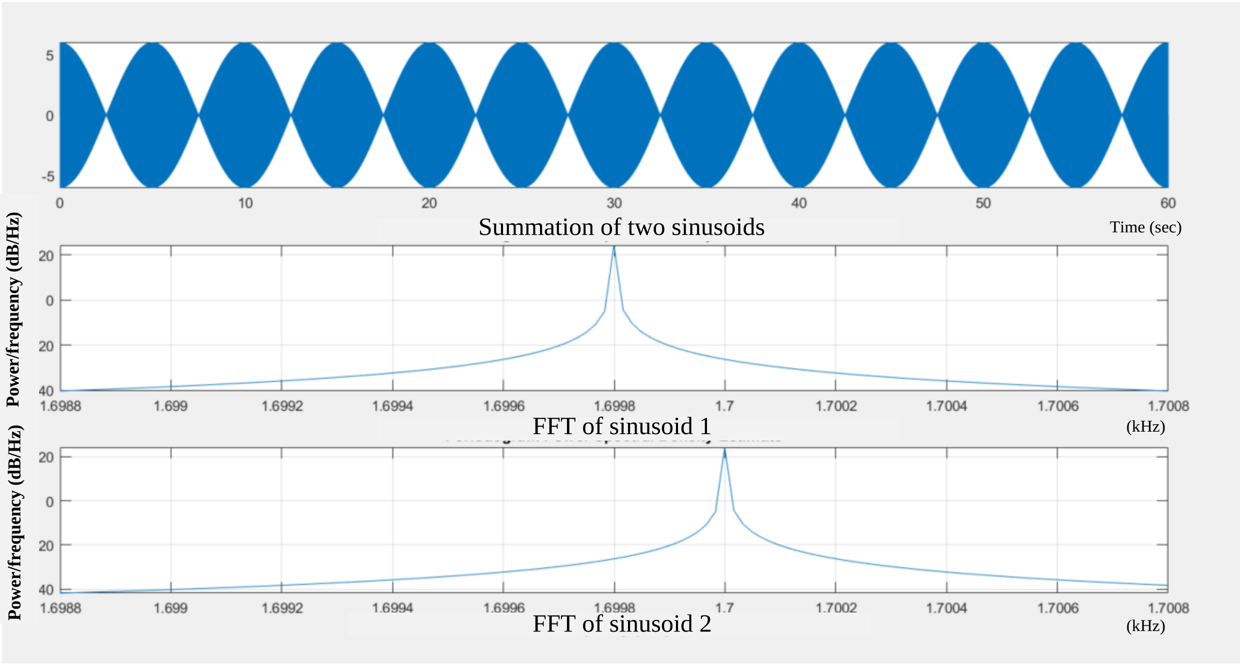


Figure 8.9 Summation simulation of 1699.8Hz and 1700Hz sinusoids

Matlab analysis as illustrated in Figure 8.9 shows that the acoustic summation of 1699.8Hz and 1700Hz introduces a modulated sinusoid with period of 5 seconds.

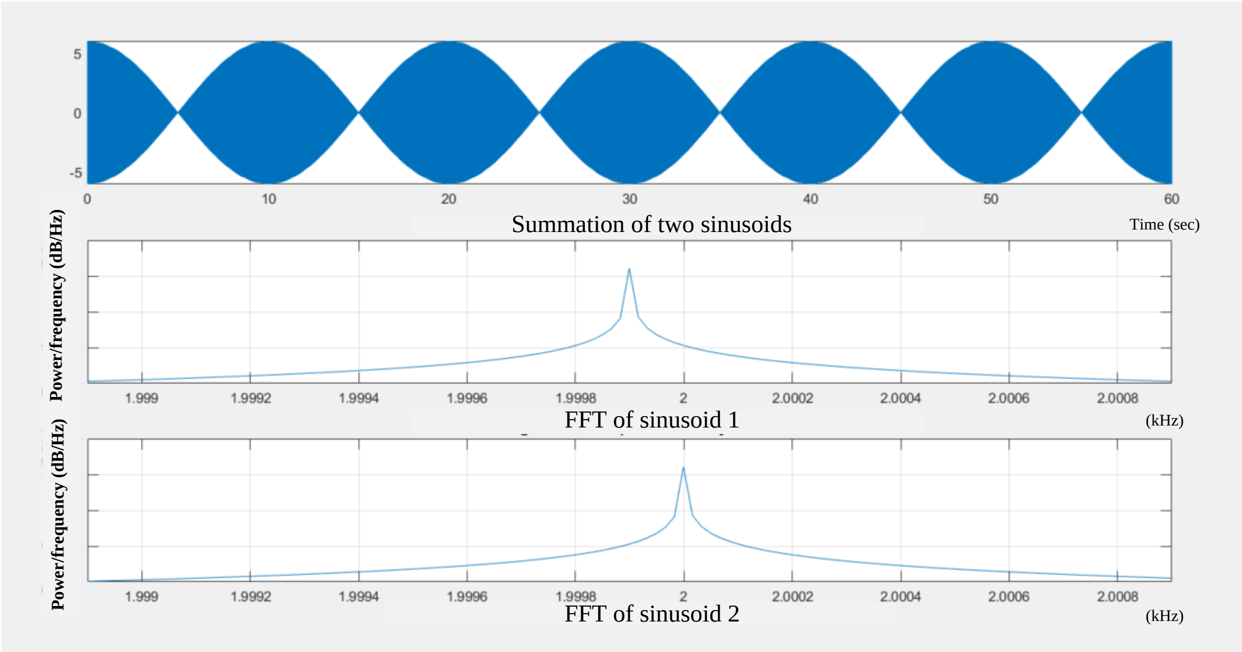


Figure 8.10 Summation simulation of 1999.9Hz and 2000Hz sinusoids

Matlab analysis as illustrated in Figure 8.10 shows that the acoustic summation of 1999.9Hz and 2000Hz introduces a modulated sinusoid with period of 10 seconds.

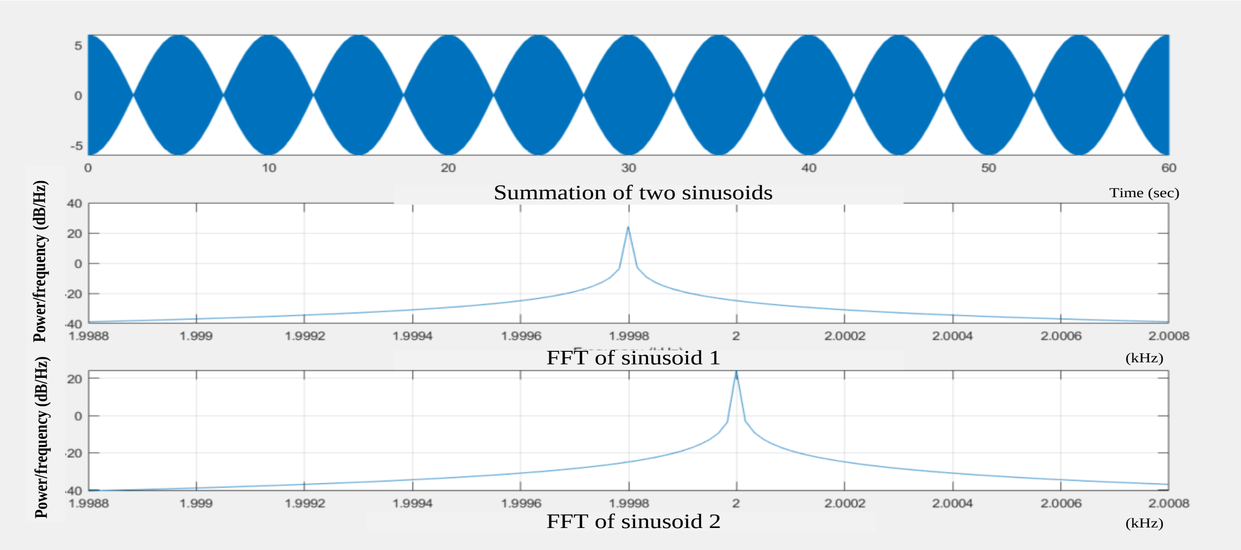


Figure 8.11 Summation simulation of 1999.8Hz and 2000Hz sinusoids

Matlab analysis as illustrated in Figure 8.11 shows that the acoustic summation of 1999.8Hz and 2000Hz introduces a modulated sinusoid with period of 5 seconds.

8.5.1.2 HFI ANC testing on the HEMAS platform

The $f_{speaker}/f_{HFI}$ frequency pairs listed in table 8.1 were then applied to the HEMAS platform and acoustic results as captured using a microphone are presented below.

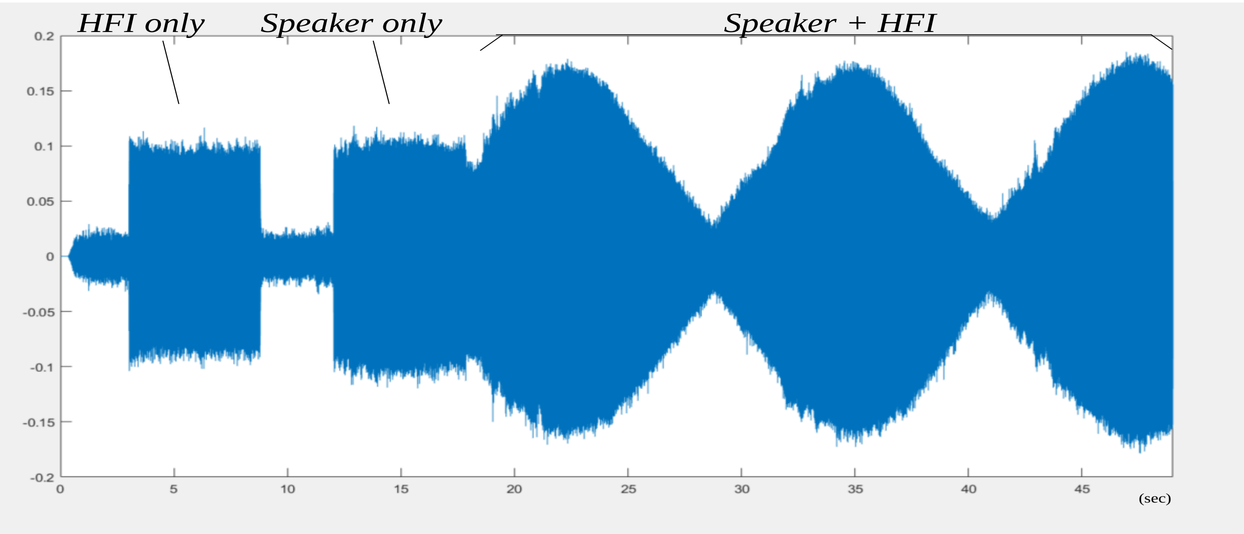


Figure 8.12 HFI ANC HEMAS platform, $f_{speaker}$ = 1399.9Hz f_{HFI}=1400Hz

Audio link 8.1: HFI ANC 1399.9Hz/1400Hz	https://youtu.be/u6LvHpmaBr4

Note that analysis of this testing is provided below this group of Figures.

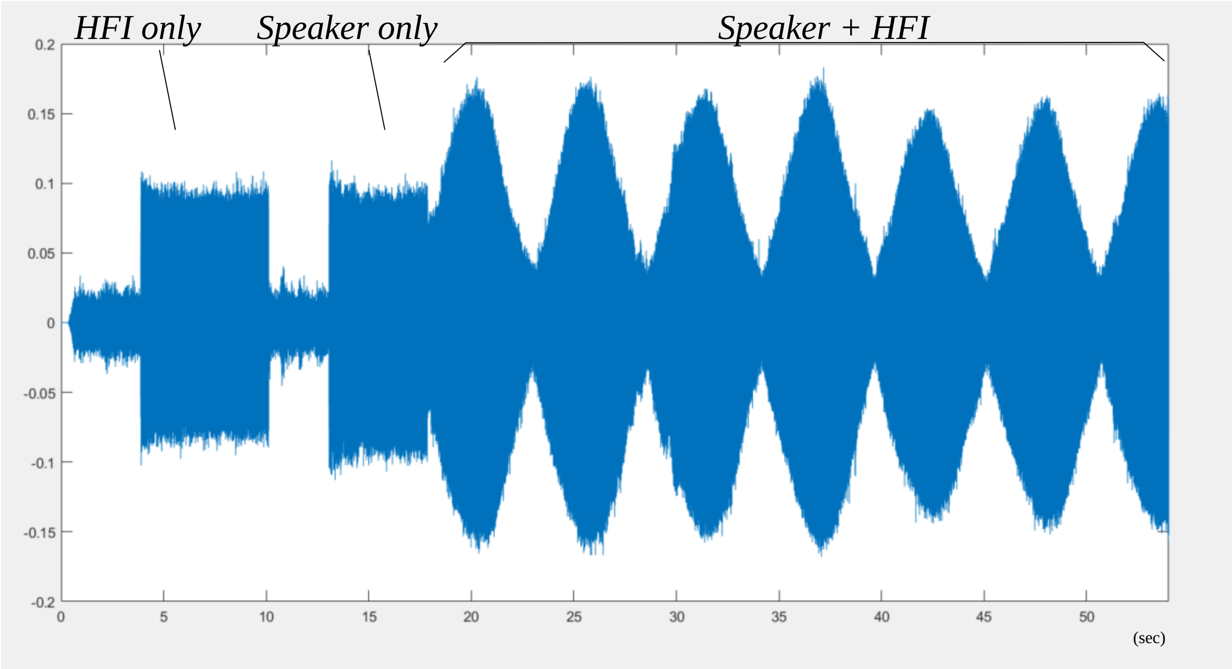


Figure 8.13 HFI ANC HEMAS platform, $f_{speaker}$ = 1399.8Hz f_{HFI}=1400Hz

Audio link 8.2: HFI ANC 1399.8Hz/1400Hz	https://youtu.be/BPzfQeg-0Is

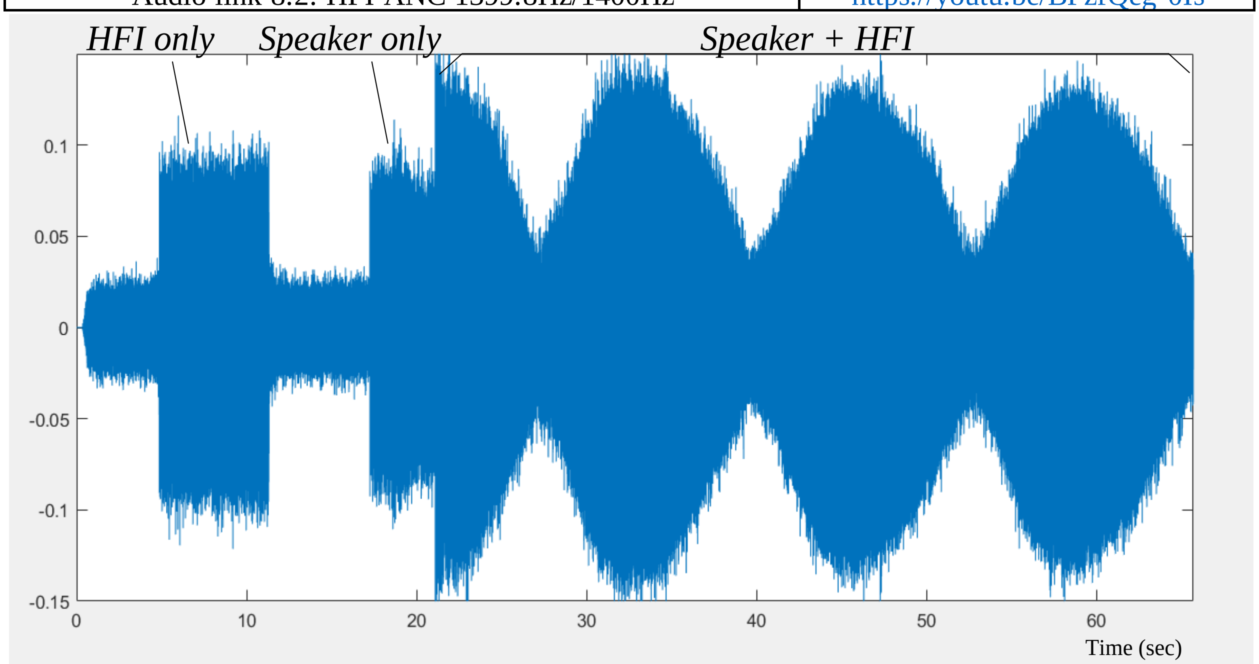


Figure 8.14 HFI ANC HEMAS platform, $f_{speaker}$ = 1499.9Hz f_{HFI}=1500Hz

Audio link 8.3: HFI ANC 1499.9Hz/1500Hz	https://youtu.be/NWSXEdrYrt0

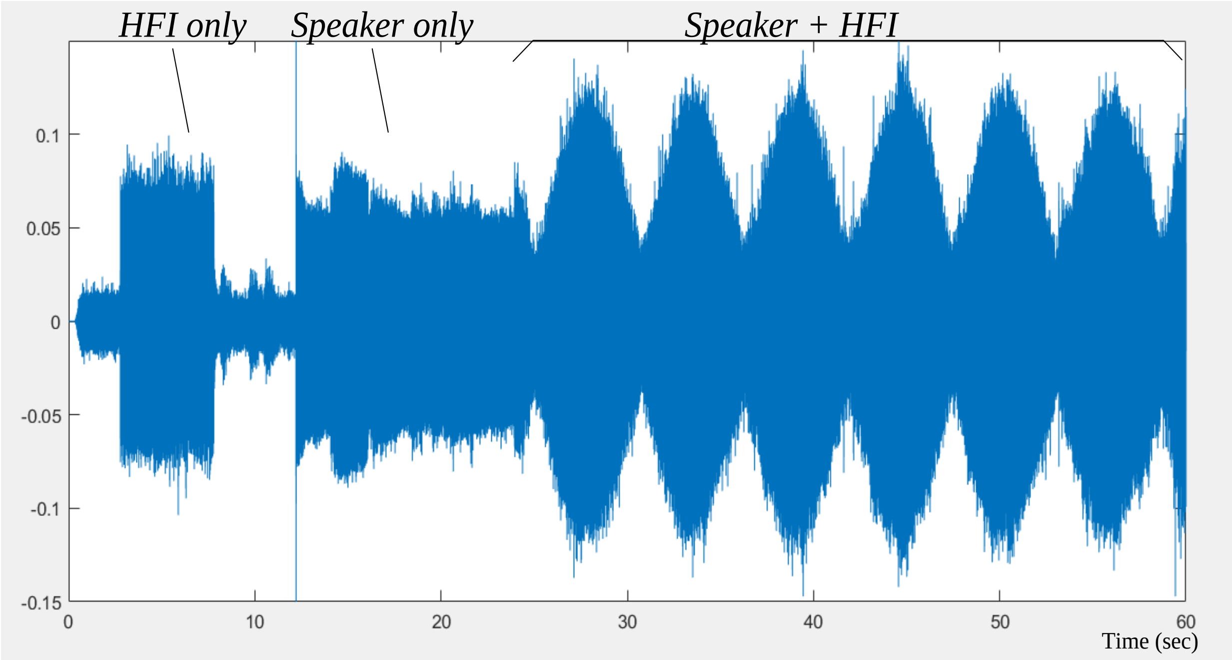


Figure 8.15 HFI ANC HEMAS platform, $f_{speaker}$ = 1499.8Hz f_{HFI}=1500Hz

Audio link 8.4: HFI ANC 1499.8Hz/1500Hz	https://youtu.be/Oi8xyQ88Dis

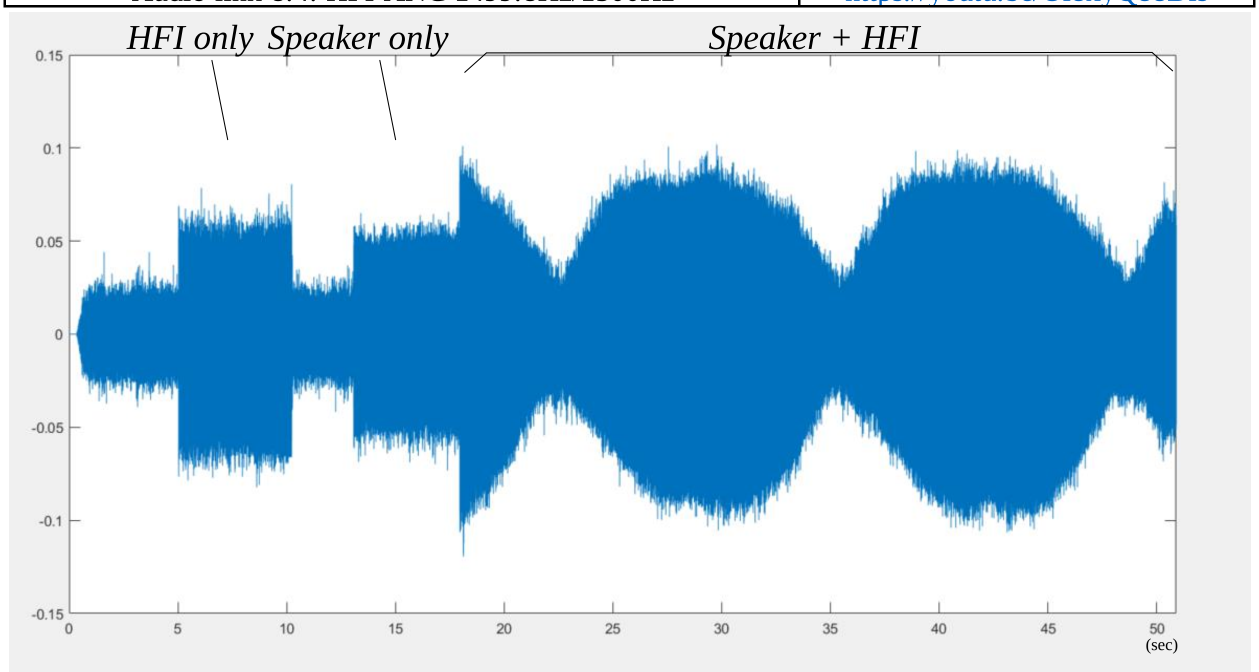


Figure 8.16 HFI ANC HEMAS platform, $f_{speaker}$ = 1699.9Hz f_{HFI}=1700Hz

Audio link 8.5: HFI ANC 1699.9Hz/1700Hz	https://youtu.be/j9XY_uJqTJM

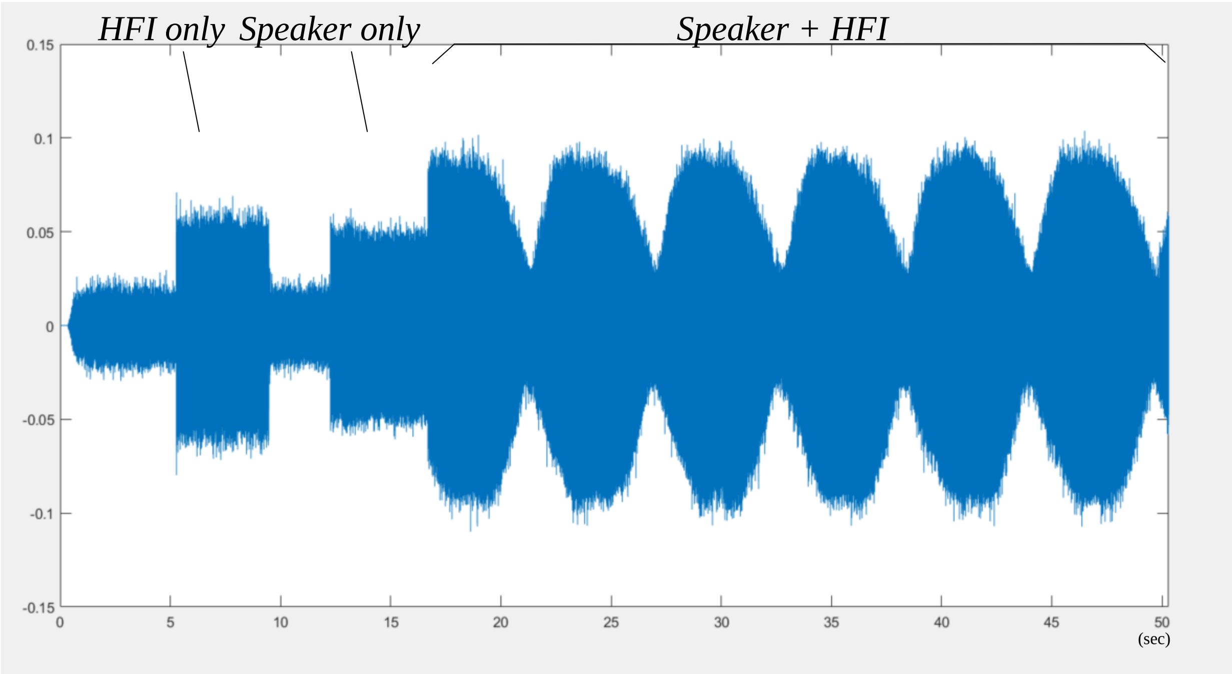


Figure 8.17 HFI ANC HEMAS platform, $f_{speaker}$ = 1699.8Hz f_{HFI}=1700Hz

Audio link 8.6: HFI ANC 1699.8Hz/1700Hz	https://youtu.be/35lnaJ2w33Q

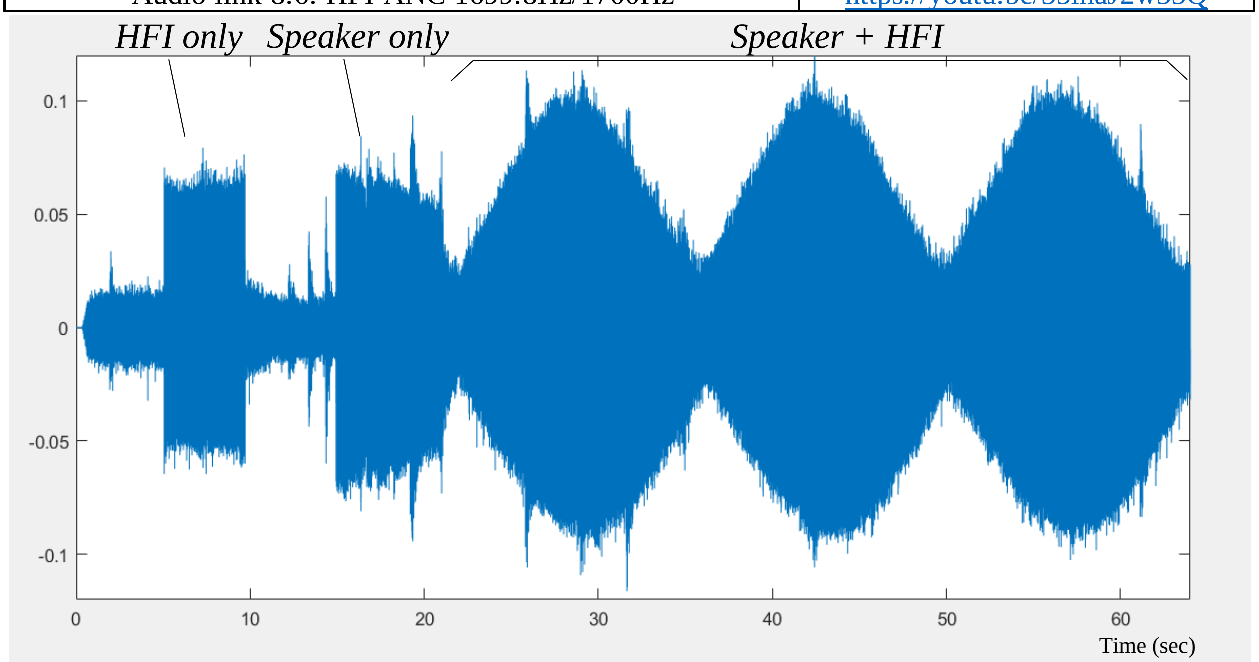


Figure 8.18 HFI ANC HEMAS platform, $f_{speaker}$ = 1999.9Hz f_{HFI}=2000Hz

Audio link 8.7: HFI ANC 1999.9Hz/2000Hz	https://youtu.be/SQJWkQmcNSY

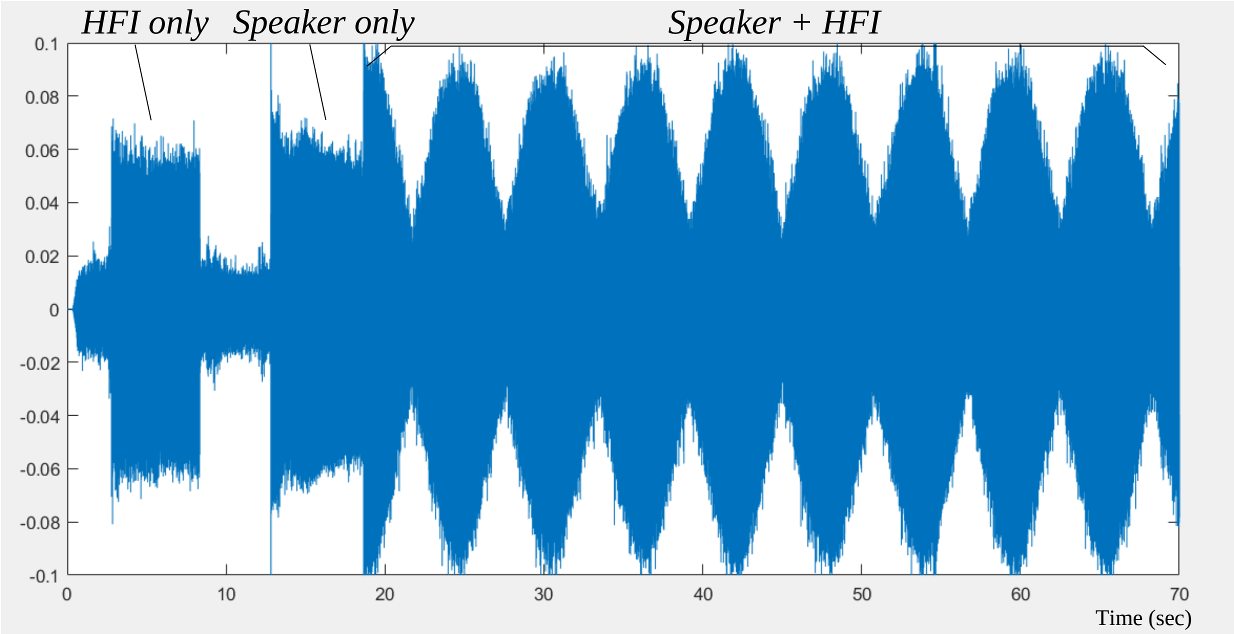


Figure 8.19 HFI ANC HEMAS platform, $f_{speaker}$ = 1999.8Hz f_{HFI}=2000Hz

Audio link 8.8: HFI ANC 1999.8Hz/2000Hz	https://youtu.be/rRO-dlbFkrM

Figures 8.12-8.19 illustrate the acoustic capturing results from the HFI ANC process for the frequency pairs set out in Table 8.1. For each test, at first only HFI is enabled and then only the speaker tonal noise aiming to illustrate and calibrate the amplitude of each of the two sound waves. Then both HFI and speaker tonal waves are enabled simultaneously allowing the acoustic cancellation to take place.

The first observation from Figures 8.12-8.19 is that the resultant noise increases and decreases over time as per Matlab simulations of Figures 8.4-8.11 illustrating that the noise cancellation process is taking place. Specifically when the two acoustic waveforms (Speaker/HEMAS HFI) are in phase their positive and negative sound waves add up resulting into high amplitude and a maximum at the resultant noise. When the two waveforms are in anti-phase the Active Noise Cancellation process takes place and the resultant noise is at its minimum amplitude. The acoustic observation of a human witnessing the above experiment that can be experienced by clicking on the web audio links, is that every few seconds the noise increases and decreases overtime in a periodic manner. One key difference though between simulations of 8.4-8.11 and hardware testing displayed in Figures 8.12-8.19 is that the resultant noise does not attenuate completely down to 0. This is firstly because there is some ambient environmental noise within the lab and secondly because HFI produces some lower amplitude harmonics. These results can be improved if the PWM frequency was increased from the current setting of 10kHz.

The noise improvement from one HFI ANC test is shown diagrammatically in Figure 8.20 and the acoustic cancellation effectiveness over the whole group of tests is summarized in Table 8.2 below.

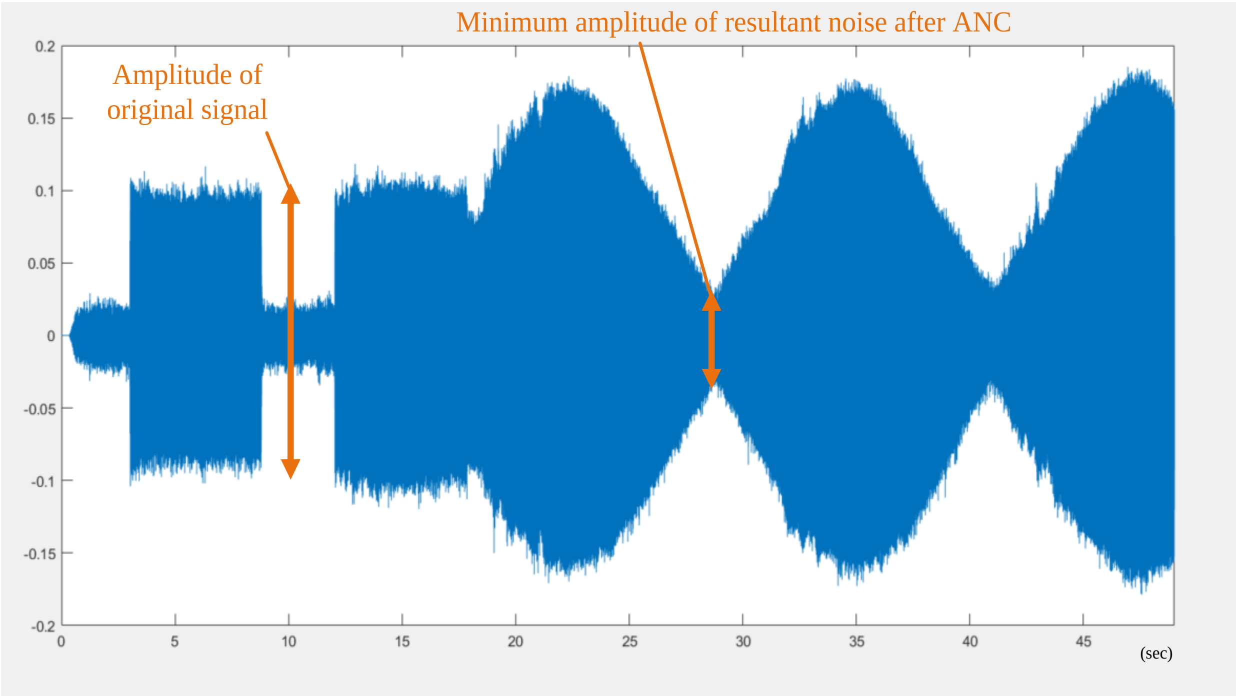


Figure 8.20 HEMAS testing ANC gain illustration

Test frequency pair	Resultant minimum noise amplitude using the HFI ANC method
1399.9/1400Hz	32% of original wave
1399.8/1400Hz	39% of original wave
1499.9/1500Hz	47% of original wave
1499.8/1500Hz	61% of original wave
1699.9/1700Hz	48% of original wave
1699.8/1700Hz	56% of original wave
1999.9/2000Hz	40% of original wave
1999.8/2000Hz	44% of original wave

Table 8.2. HFI ANC Noise reduction evaluation

A varying degree of acoustic noise reduction is observed ranging between 32-61%. This variation is primarily because the tuning of the amplitude of the two waveforms (HFI/speaker) was manual. If the two waveforms were matched to have exactly the same amplitude in an automated manner the cancellation process would be more effective and results would improve considerably. While the purpose of this experiment is to illustrate the proposed concept of ANC by means of HFI, an actual product would need to function in a closed loop manner processing the noise and calculating the anti-noise in a wide band spectrum. This closed loop wide band function has thus been added in the Future work of this thesis.

Looking at the data in Figures 8.12-8.19 more closely, one additional observation is that the sliding period is higher than the expected 10 and 5 seconds of Matlab simulations 8.4-8.11. This additional delay is consistent with the playback laptop and the HEMAS hardware being relatively misaligned on measuring time by an error of about 8ns per sinusoidal cycle. This delay is due to oscillator/PLL drift of the two acoustic systems (HEMAS / laptop). Such a frequency drift is typically encountered on oscillator components upon aging or upon exposure to temperature variations. However this frequency deviation would have exceptionally small impact on the ANC process as at the point that the two cancelling sound waves are in antiphase relation and the ANC is active, 8ns error over a sinusoidal period of 714286ns for the 1.4kHz injection is negligeable.

8.5.2 Experiment 2: HFI acoustic versatility evaluation

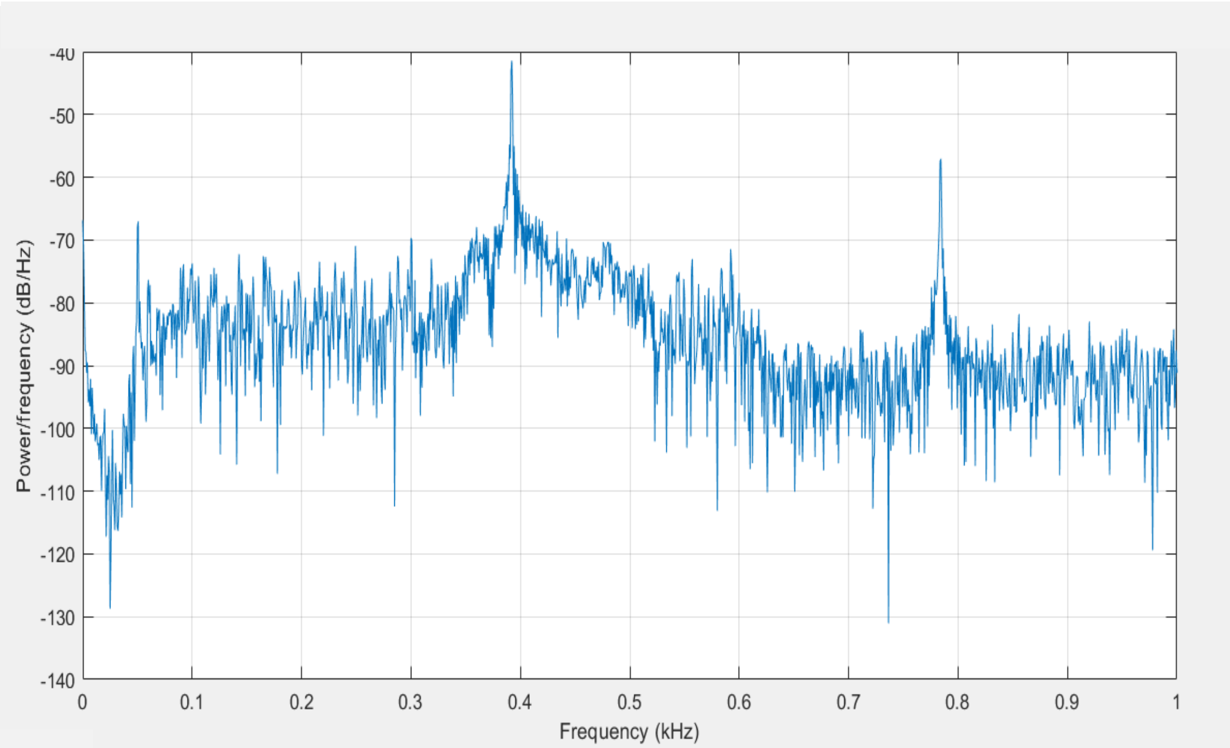
Section 8.5.1 demonstrated that the HFI ANC method can be used towards reducing the acoustic noise within a system. However, one possible question that could be posed is whether a motor drive is versatile enough to generate acoustic waves. Aiming to demonstrate the acoustic flexibility of the method in a creative manner, HFI is set up so that the HEMAS platform acts as a musical instrument. A sequence of musical notes from the recognisable classical masterpiece "Eine Kleine Nachtmusic" by Mozart is produced by means of HFI.

The motor's digital controller's implementation code was thus updated with an array of note frequencies and durations so as to synthesize the selected classical song. Note that the implementation code for the HFI notes of the song can be found in Appendix B.2.3 of this thesis. The song "Eine Kleine Nachtmusic" generated by HFI on the HEMAS platform as captured using a microphone can thus be found in the web links below.

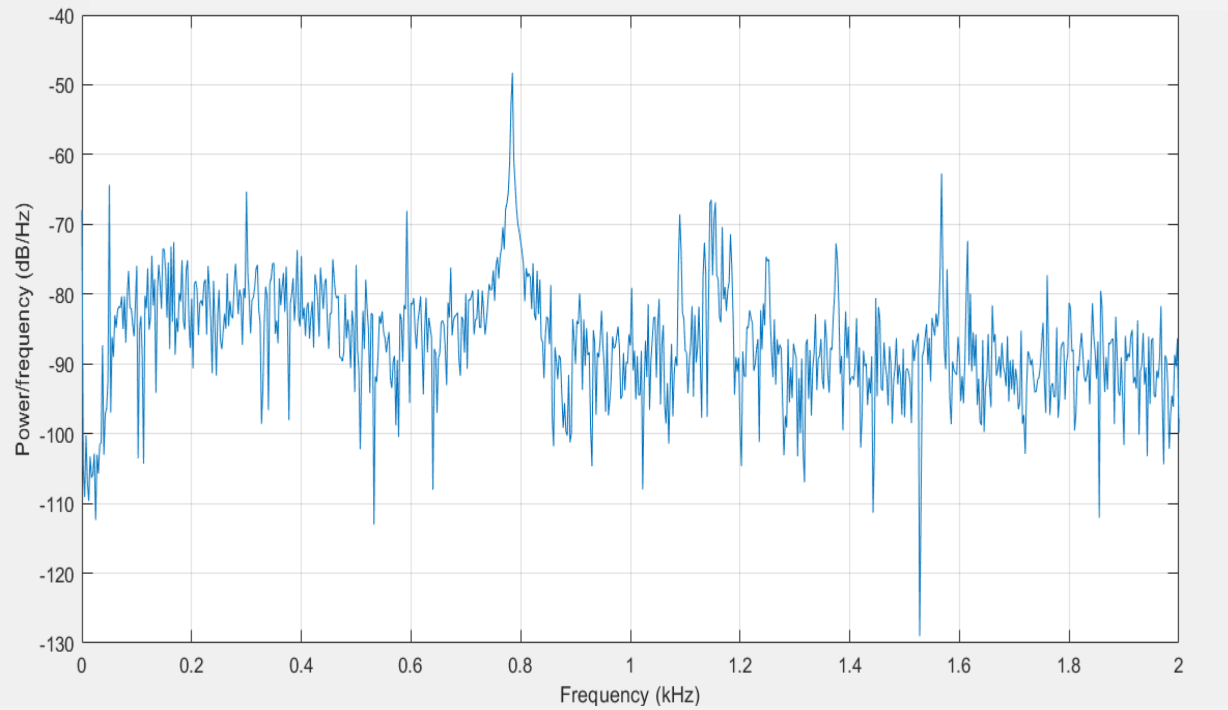
Audio link 8.9: HEMAS HFI song Eine Kleine Octave4	https://youtu.be/Y2mP2A0oCtA
Audio link 8.10: HEMAS HFI song Eine Kleine Octave5	https://youtu.be/P0QAEPoSzbM

Listening to the song produced by the HEMAS motor drive and taking under consideration that the motor was never designed to produce acoustic waves is of significant interest illustrating the acoustic capabilities of a motor drive.
In terms of acoustic data analysis, the FFT analysis of two HFI notes is shown in Figures 8.21 and 8.22 illustrating a portion of the acoustic frequency range of the method.

<u>HEMAS HFI musical note G octave 4 and 5</u>



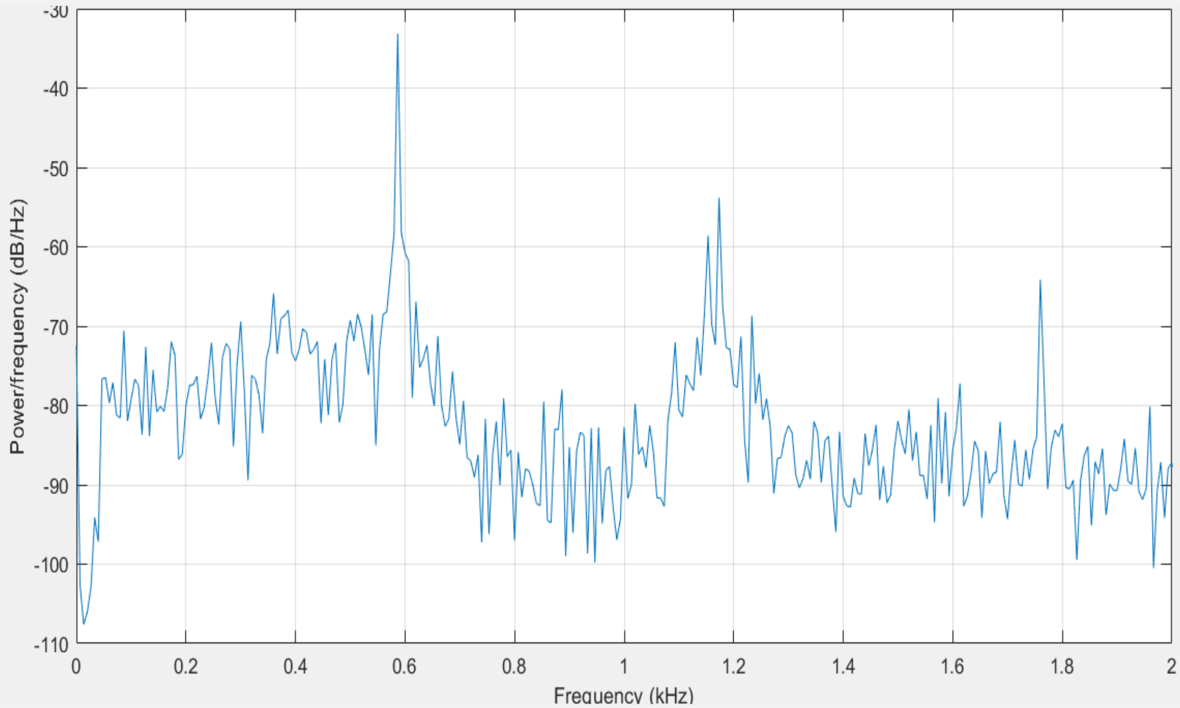
FFT of HFI note G Octave 4 frequency 392Hz



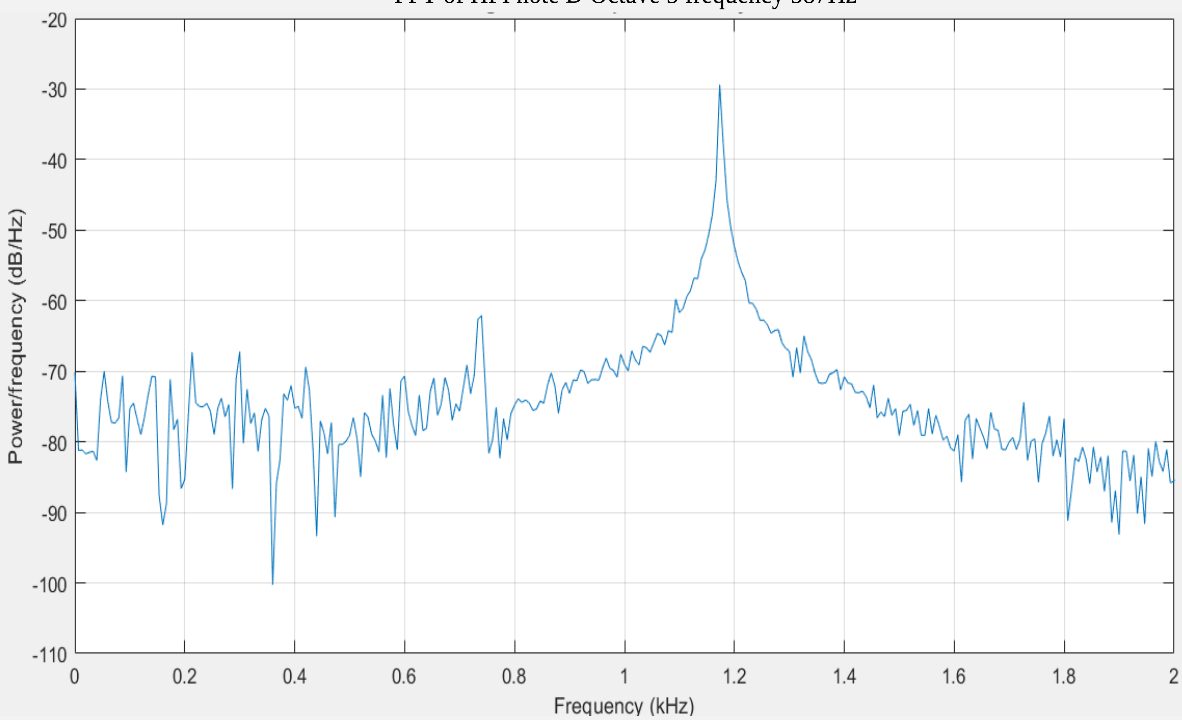
FFT of HFI note G Octave 5 frequency 784Hz

Figure 8.21 FFT of note G octave 4 and 5 from HFI

<u>HEMAS HFI musical note D octave 5 and 6</u>



FFT of HFI note D Octave 5 frequency 587Hz



FFT of HFI note D Octave 6 frequency 1175Hz

Figure 8.22 FFT of note D octave 5 and 6 from HFI

8.6 Analysis of test results

Experiment 1 demonstrated that the HFI ANC method when applied to tonal noise can achieve tonal noise cancellation resulting into 32%-61% of the original noise amplitude. These results are very encouraging taking under consideration that the HEMAS drive was never designed to act as an acoustic noise generator. The method shows that the motor can thus achieve two functions, spinning and enabling a quieter overall acoustic system.

Experiment 2 went one step further and attempted to illustrate that the HEMAS motor drive and to an extent any drive can be seen as a versatile acoustic generation tool. Listening to the Eine Kleine Nachtmusic song produced by applying HFI to the HEMAS platform is illustrating the flexibility of the proposed method.

8.7 Conclusions

This chapter presented the HFI ANC method, a novel scheme that uses the existing electrical and mechanical infrastructure of a motor drive to cancel existing noise residing within a system. It first presented the theoretical framework of the method, followed by analysis and experimental testing on the Helicopter Electro-Mechanical Actuation System (HEMAS). Experiment 1 showed that acoustic improvements can be obtained from the proposed method resulting into 32%-61% of original noise. This improvement is considered substantial taking under consideration that the PWM switching frequency is set to 10kHz and the digital/analog technology used is more than a decade old. The acoustic gains can be amplified if the PWM switching frequency is increased. Experiment 2 then demonstrated the versatility of the method being able to produce sound waves over a wide range of musical frequencies and octaves. Taking under consideration the results and the increased levels of electrification in the aerospace, land and marine sectors the proposed method can have a significant impact on the acoustics of systems deploying motor drives.

Chapter 9 Conclusions and Future Work

9.1 Introduction

The main objective of this research effort outlined in the abstract of this thesis was to investigate two key areas system availability and acoustics. The first task was therefore to explore and propose sensorless algorithms that can be used to take over motor control in case of a resolver failure therefore enhancing system availability. The second task was to investigate methods that improve acoustics of motor drives. Reliability, availability and acoustics are all key areas in a number of industries especially aerospace.

With regards to the availability objective, two sensorless methods were presented a Model Based Observer that is suitable for medium to high speed and a Saliency Based Method best suited for standstill and low speed.

With regards to the acoustics objective, the thesis first presented a variant of the Pseudo Random High Frequency Injection (PRHFI) algorithm aiming to reduce the perception of HFI acoustic noise and advance the current state of the art in this area. The second acoustic improvement algorithm and perhaps the most novel and promising research contribution presented in this thesis is the Active Noise Cancellation (ANC) by means of High Frequency Injection (HFI). HFI ANC showed effectiveness towards noise cancellation and was successful in adding an acoustic capability feature to an otherwise conventional motor drive.

The section below provides more details on the outcomes of each of the research areas investigated. A summary of the project's achievements and ideas for further work are then presented in sections 9.4 and 9.5 respectively.

9.2 Investigation of sensorless methods as means of enhancing availability

Chapter 2 introduced the architecture and reliability issues associated with resolvers and therefore the motivation towards the use of sensorless methods towards enhancing system availability. It then presented a survey of sensorless methods aiming to identify the features, advantages and disadvantages of each method. Two sensorless methods were selected out of this review activity to be investigated further and tested on the targeted HEMAS platform. First a Back EMF model based observer suitable for medium to high speed that was simulated implemented and tested on the HEMAS platform as detailed Chapter 5 of this thesis. The model based observer was shown to be able to track the rotor angle and speed when compared to the resolver readings. Then a saliency based observer applying stator iron saturation suitable for standstill or low speed was simulated and tested on the HEMAS platform as detailed in Chapter 6. The saliency observer was also able to track the resolver readings. Taking into account that the two sensorless methods can cover a wide speed range from standstill to the maximum speed supported by the test rig it is considered that the two implemented algorithms provide a comprehensive method towards replacing the resolver in case of a failure and enhancing system availability.

9.3 Optimizing acoustics using PRHFI and the HFI ANC method

Chapter 3 provided an introduction to the science of acoustics and an analysis of aircraft acoustic noise sources. This literature review provided the setting, background and applicability of the proposed acoustic improvement algorithms. Chapter 7 proposed a variant of the Pseudo Random High Frequency Injection (PRHFI) algorithm aiming to advance the current state of the art in the area. It provided analysis and test results illustrating acoustic improvements towards reducing the perception of HFI acoustic noise. Chapter 8 followed a substantially more novel and innovative route on acoustic noise from HFI. Instead of considering this noise as a negative byproduct it attempted to recycle and harvest this sound towards a useful function and specifically achieving Active Noise Cancellation within a system. ANC by means of HFI was very effective and was able to reduce to 32%-61% of the original noise amplitude. Taking under consideration the increased level of electrification in a number of industrial sectors, adding an acoustic generation capability to an otherwise conventional motor drive using analog and digital technology that has been around for decades proves a concept that can lead to a possibly very promising technological advancement.

9.4 Summary of project achievements

The achievements of this work can be summarised to the below:

- Analysis of sensorless control methods that can be used towards system availability of motor drives in case of resolver failures.
- Analysis, simulation, implementation and testing of a model based observer and a saliency based observer providing a comprehensive motor control method and availability improvement throughout a wide speed range. During the saliency method investigation, a novel saliency tracking algorithm concept was also proposed in section 6.11.
- Analysis, simulation and testing of the proposed Pseudo Random High Frequency Injection variant aiming to reduce the perception of HFI acoustic noise.
- Perhaps the most significant and novel research contribution in this thesis is the proposed ANC by means of HFI. The proposed method showed effectiveness towards achieving noise cancellation using a conventional motor drive reducing acoustic noise to 32%-61% of the original waveform amplitude. The experimental tests performed also illustrated that a motor drive can function as a very versatile acoustic generation tool.

9.5 Future Work

The work performed in this thesis can be extended and expanded as listed below:

- The model based and saliency sensorless methods act as observers and their estimated angle and speed is compared with the resolver readings for analysis and evaluation purposes. One future step is to use the sensorless position/speed to feed into the Clarke/Park transforms and the speed control loop.
- The response mechanism to a resolver failure can also be investigated by means of disconnecting the resolver while the motor is spinning and allow switching and take over to sensorless control algorithms.
- The novel saliency tracking algorithm introduced in section 6.11 can be implemented in the future.
- The testing of the PRHFI algorithm in Chapter 7 focuses only on acoustics. The saliency rotor tracking capability was only tested for constant frequency HFI in Chapter 6. The PRHFI position/speed tracking can be enabled in a future project.
- The Active Noise Cancellation by means of HFI in Chapter 8 is performed for tonal noise and in open loop manner. While tonal noise is encountered in Aircraft applications [44], [45], and the method can be applied in open loop when the noise is predictable, using a microphone and applying the method in closed loop would allow the method to be expanded to cover a mixture of tonal and wideband noise. Extending the method to function in closed loop and covering wideband noise would be a natural continuation enabling the method to be more widely applied in commercial products.

References

[1] A. Boglietti, A. Cavagnino, A. Tenconi, S. Vaschetto, "The Safety Critical Electric Machines and Drives in the More Electric Aircraft: a Survey", 35th Annual Conference of IEEE Industrial Electronics, 2009.

[2] Pat Wheeler “Technology for the more and all electric aircraft of the future”, 2016 IEEE International Conference on Automatica (ICA-ACCA), 2016.

[3] John W. Bennett, Glynn J. Atkinson, Barrie C. Mecrow, David J. Atkinson "Fault-Tolerant Design Considerations and Control Strategies for Aerospace Drives", IEEE Transactions on Industrial Electronics, 2012.

[4] Slobodan N. Vukosavic “Electrical Machines”, Power Electronics and Power Systems, Springer, 2013.

[5] Fouad Giri “AC Electric Motors”, Advanced Design Techniques and Applications, Wiley, 2013.

[6] RTCA/DO-254 Design Assurance Guidance For Airborne Electronic Hardware, RTCA, April 19 2000.

[7] EASA CM-SWCEH-001, Development Assurance of Airborne Electronic Hardware, Issue No. 01, Revision 01.

[8] Austin Hughes “Electric Motors and Drives”, Fundamentals Types and Applications, Third Edition, Newnes, 2006.

[9] Austin Hughes, Bill Drury “Electric Motors and Drives”, Fundamentals Types and Applications, Fourth Edition, Newnes, 2012.

[10] Merve Yildirim, Mehmet Polat, Hasan Kürüm “A survey on comparison of electric motor types and drives used for electric vehicles”, 2014 16th International Power Electronics and Motion Control Conference and Exposition, 2014.

[11] C. Vangsness “Comparison of brush and brushless servo motor designs” IEEE Conference Record of 1988 Fortieth Annual Conference of Electrical Engineering Problems in the Rubber and Plastics Industries.

[12] Dancho Ivanov, Ivan Maradzhiev, Tsvetana Grigorova "FPGA Implementation of Microstepping Control of Stepper Motor with Advanced Mixed Current Decay", 2021 12th National Conference with International Participation (ELECTRONICA), 2021.

[13] Ningli Wang, Hao Feng, "Research on Torque Ripple and its Minimization Method of Switched Reluctance Motor", 2018 5th International Conference on Information, Cybernetics, and Computational Social Systems (ICCSS), 2018.

[14] Dejan Kos, Aljaz Kapun, Milan Curkovic, Karel Jezernik, "Efficient Stepper Motor Torque Ripple Minimization Based on FPGA Hardware Implementation", IECON 2006 - 32nd Annual Conference on IEEE Industrial Electronics, 2006.

[15] Aiyuan Wang, Zhihao Ling "Improved Design for Reduction of Torque Ripple of Brushless DC Motor", 2009 International Conference on Industrial and Information Systems, 2009.

[16] V. Viswanathan, S. Jeevananthan "Torque ripple comparison of a two-level and a three-level inverter fed brushless DC motor drives at different switching frequency", 2014 International

Conference on Green Computing Communication and Electrical Engineering (ICGCCEE), 2014.

[17] S. Sakunthala, R. Kiranmayi, P. Nagaraju Mandadi “A study on industrial motor drives: Comparison and applications of PMSM and BLDC motor drives”, 2017.

[18] L. Papini, P. Connor, C. Pately, L. Empringham, C. Gerada, P. Wheeler "Design And Testing of Electromechanical Actuator for Aerospace Applications", 2018 25th International Workshop on Electric Drives: Optimization in Control of Electric Drives (IWED), 2018.

[19] M. Rottach, C. Gerada, P.W. Wheeler "Design Optimisation of a Fault-Tolerant PM Motor Drive for an Aerospace Actuation Application", 7th IET International Conference on Power Electronics, Machines and Drives (PEMD 2014), 2014.

[20] Michael Rottach "Drive-System Optimization for a Helicopter Electromechanical Actuation System, Ph.D Thesis, University of Nottingham, 2014.

[21] Shaohua Chen, Yunlong Zhao, Haitao Qiu, Xiaojun Ren "High-Precision Rotor Position Correction Strategy for High-Speed Permanent Magnet Synchronous Motor Based on Resolver", IEEE Transactions on Power Electronics, 2020.

[22] Sukanta Halder, Anubhav Agrawal, Pramod Agarwal, S.P. Srivastava "Resolver based position estimation of vector controlled PMSM drive fed by matrix converter", 2016 Second International Innovative Applications of Computational Intelligence on Power, Energy and Controls with their Impact on Humanity (CIPECH), 2016.

[23] Muhammed Fazlur Rahman , Sanjeet K. Dwivedi, “Modeling, simulation and control of electrical drives”, October 2019, Publisher: Institution of Engineering and Technology.

[24] R. Dhaouadi, N. Mohan and L. Norum, "Design and implementation of an Extended Kalman filter for state estimation of a permanent magnet synchronous motor’, IEEE Trans. Power. Electron., vol. 6, pp. 491–497, 1991.

[25] W. Chongwu, H. Yuyao, L. Hong “The study on the PMSM sensorless control using the sub-optimal fading extend Kalman filter”, 2013 IEEE 10th International Conference on Power Electronics and Drive Systems (PEDS).

[26] C. Moon, Kee H. Nam, M. K. Jung, C. H. Chae, Y. A. Kwon, “Sensorless speed control of permanent magnet synchronous motor using Unscented Kalman Filter “ 2012 Proceedings of SICE Annual Conference (SICE).

[27] H. Li, Z. Wang “Sensorless Control for PMSM Drives Using the Cubature Kalman Filter based Speed and Flux Observer” 2018 IEEE International Conference on Electrical Systems for Aircraft, Railway, Ship Propulsion and Road Vehicles & International Transportation Electrification Conference (ESARS-ITEC).

[28] C. Varghese, V Suresh, “Control of Permanent Magnet Synchronous Motor using MRAS”, International Journal of Latest Trends in Engineering and Technology, Vol. 3, Issue 4, pp. 71-77, March 2014.

[29] F. Chen, X. Jiang, X. Ding, C. Lin, "FPGA-based sensorless PMSM speed control using adaptive sliding mode observer", IECON 2017 - 43rd Annual Conference of the IEEE Industrial Electronics Society.

[30] Z. Baozhou, W. Lili, L. Aiying, L. Fuqiang, Y. Chao, D. Jianliang, "Study of a New Detection Technology for Sensorless PMSM Drive with Flux Linkage Observer", 2011 International Conference on Consumer Electronics, Communications and Networks.

[31]Y. Jiang, W. Xu, C. Mu, "Improved SOIFO-based rotor flux observer for PMSM sensorless control", IECON 2017 - 43rd Annual Conference of the IEEE Industrial Electronics Society.

[32]A. Accetta, M. Cirrincione, M. Pucci “Sensorless Control of PMSM by a Linear Neural Network: TLS EXIN Neuron” IECON 2010 - 36th Annual Conference on IEEE Industrial Electronics Society.

[33]J. Moreira and T. A. Lipo, “Modeling saturated AC machines including air gap flux harmonic components,” IEEE Trans. Ind. Appl., vol. 28, no. 2, pp. 343–349, Mar./Apr. 1992.

[34]J.M. Liu, Z.Q. Zhu “Sensorless control based on third harmonic back-EMF and PLL for permanent magnet synchronous machine“, 2013 International Conference on Electrical Machines and Systems (ICEMS).

[35]S. Medjmadj, D. Diallo, M. Mostefai, C. Delpha, and A. Arias, “PMSM drive position estimation: contribution to the high frequency injection voltage selection issue”, IEEE Trans. Energy Convers., vol. 30, no. 1, pp. 349-358, 2015.

[36]R. Ni, K. Lu, F. Blaabjerg, D. Xu , “A comparative study on pulse sinusoidal high frequency voltage injection and INFORM methods for PMSM position sensorless control” IECON 2016 - 42nd Annual Conference of the IEEE Industrial Electronics Society.

[37]C. Caruana, G. M. Asher and J. Clare, “Sensorless Flux Position Estimation at Low and Zero Frequency by measuring Zero-Sequence Current in Delta Connected Cage Induction Machines”, 38th IAS Annual Meeting on Conference Record of the Industry Applications Conference, 2003.

[38]Y. Iwaji, R. Takahata T. Suzuki, M. Hano "All-Speed-Range Drive for Surface Permanent Magnet Synchronous Motors by Combined Use of Two Position Sensorless Methods" 2014 IEEE 5th International Symposium on Sensorless Control for Electrical Drives.

[39]H. Tang ,H. Li ,J. Lin "Research on Sensorless Control Method of PMSM Based on a Kalman Filter Sliding Mode Observer, 2018 10th International Conference on Measuring Technology and Mechatronics Automation (ICMTMA), IEEE, 2018.

[40]Di Rito, G. Galatolo, R. Schettini, F.: Self-monitoring electro-mechanical actuator for medium altitude long endurance unmanned aerial vehicle flight controls. Adv. Mech. Eng. 8(5), 1687–8140 (2016). https://doi.org/10.1177/1687814016644576

[41]M. Mazzoleni, G. Di Rito, F. Previdi, "Electro-Mechanical Actuators for the More Electric Aircraft", Springer, Berlin (2021), https://doi.org/10.1007/978-3-030-61799-8

[42]K. Sahri, M. Pietrzak-David, M. Fadel, A. Kheloui, "Sensorless Tolerant Fault Control for Dual Permanent Magnet Synchronous Motor Drive with Global FPGA Emulator", 2018 20th European Conference on Power Electronics and Applications (EPE'18 ECCE Europe), 2018.

[43]Paul E. Sabine “Acoustics and architecture”, McGraw-Hill Book Company Inc, 1932.

[44]M. Haddaoui " Development of a Propeller Source Noise Model", MSc thesis, Delft University of Technology, 2019.

[45]R. J. Astley, "“Predicting and Reducing Aircraft Noise"”, ICSV14, 14th International Congress on Sound Vibration, Cairns, Australia, 2007.

[46]Ahmed F. El-Sayed "Aircraft Propulsion and Gas Turbines, second edition", Taylor & Francis, CRC Press, 2017.

[47]Credit for piston propeller picture,

https://hangar.flights/airplanes/basics-and-types-of-airplane-propellers/

[48] Credit for piston propeller picture, http://www.pilotfriend.com/aero_engines/aero_pst_eng.htm

[49] Credit for turbo jet engine picture, https://en.wikipedia.org/wiki/Junkers_Jumo_004

[50] Credit for turbo jet engine picture, https://www.century-of-flight.net/turbojet-vs-turbofan-explained/

[51] Credit for photo of turbofan engine photo on a Boeing 787 (Benoit Tessier / Reuters) https://www.theatlantic.com/technology/archive/2018/02/engine-failure/552959/

[52] Credit for Turbofan block diagram https://en.wikipedia.org/wiki/Turbofan

[53] Credit for Turboprop picture https://aerocorner.com/blog/types-of-aircraft-engines/

[54] Credit for Turboprop diagram https://en.wikipedia.org/wiki/Turboprop

[55] Credit for Turboshaft picture https://aerocorner.com/blog/types-of-aircraft-engines/

[56] Credit for Turboshaft diagram https://en.wikipedia.org/wiki/Turboshaft

[57] Credit for SR-72 picture https://en.wikipedia.org/wiki/Lockheed_Martin_SR-72

[58] Credit for Scramjet engine diagram https://www.clearias.com/scramjet-engine/

[59] Credit for Ramjet engine diagram https://aerospacenotes.com/propulsion-2/ramjet-engine/

[60] Yong Li, Malcolm Smith, Xin Zhang, Nicolas Molin "Noise Sources Control of an Aircraft Landing Gear", 13th AIAA/CEAS Aeroacoustics Conference (28th AIAA Aeroacoustics Conference), 2007.

[61] B. Magliozzi, D. B. Hanson, and R. Amiet, "Aeroacoustics of Flight Vehicules: Theory and Practice", NASA Reference Publication 1258, Vol. 1 WRDC Technical Report 90-3052 1 (1991).

[62] Hui Jiang Audible noise reduction in the high frequency injection based sensorless torque control for EPS applications, Ph.D thesis, University of Nottingham, 2012.

[63] G. Wang, G. Zhang, D. Xu, "Position Sensorless Control Techniques for Permanent Magnet Synchronous Machine Drives", Springer December 2019.

[64] X. Luo, Q. Tang, A. Shen, and Q. Zhang "PMSM Sensorless Control by Injecting HF Pulsating Carrier Signal Into Estimated Fixed-Frequency Rotating Reference Frame", IEEE transactions on industrial electronics, vol. 63, no. 4, april 2016.

[65] J. Radosz, "Effects of Tonal Noise on Workers' Annoyance and Performance", Noise Health. 2021 Oct-Dec; 23(111): 117–127, doi: 10.4103/nah.NAH_28_20.

[66] AgustaWestland is currently part of the Leonardo S.p.A under its helicopters division https://www.leonardo.com/en/home

[67] M. Christmann, S. Seemann, and P. J¨anker, "Innovative approaches to electromechanical flight control actuators and systems", Recent Advances in Aerospace Actuation Systems and Components, pp. 17–23, 2010.

[68] C. xiong Pan, J. zhou Zhang, L. feng Ren, and Y. Shan, "Effects of rotor downwash on exhaust plume flow and helicopter infrared signature," Applied Thermal Engineering, vol. 65, no. 1, pp. 135 – 149, 2014.

[69] D. Banerjee, Z. Li, S. Mukundan, A. Balamurali, C. Lai, N. C. Kar, "Online Parameter Estimation and Self Commissioning of Permanent Magnet Motor Drive", 2018 21st International Conference on Electrical Machines and Systems (ICEMS), 2018.

[70] H. Takahashi, H. Obara, Y. Fujimoto, "Dead time compensation for three-level flying capacitor inverter with phase shift PWM", 2018 IEEE 15th International Workshop on Advanced Motion Control (AMC), 2018.

[71] F. Mandrile, S. Musumeci, M. Palma, "Dead Time Management in GaN Based Three-Phase Motor Drives", 2021 23rd European Conference on Power Electronics and Applications (EPE'21 ECCE Europe), 2021.

[72] Yue Zhao, Zhe Zhang, Cong Ma, Wei Qiao, Liyan Qu "Sensorless Control of Surface-Mounted Permanent-Magnet Synchronous Machines for Low-Speed Operation Based on High-Frequency Square-Wave Voltage Injection", 2013 IEEE Industry Applications Society Annual Meeting, 2013.

[73] Zedong Zheng, Yongdong Li, Xi Xiao, Maurice Fadel "Mechanical Sensorless Control of SPMSM Based on HF Signal Injection and Kalman Filter", 2008 International Conference on Electrical Machines and Systems, 2018.

[74] G. Wang, R. Yang, Y. Wang, Y. Yu, and D. Xu, "Initial Rotor Position Estimation for Sensorless Interior PMSM with Signal Injection", 2010 International Power Electronics Conference, 2010.

[75] Sungmin Kim, Seung-Ki Sul "High performance position sensorless control using rotating voltage signal injection in IPMSM", Proceedings of the 2011 14th European Conference on Power Electronics and Applications, 2011.

[76] Xingwei Zhou, Bo Zhou, Feng Wu, Li Zhang, Kaimiao Wang "Analysis of the Initial Alignment Process for Doubly Salient Electromagnetic Machine Position Sensorless Startup", 22nd International Conference on Electrical Machines and Systems (ICEMS), 2019.

[77] Bangcheng Han, Yangyang Shi, Xinda Song, Kun Hong, Kun Mao "Initial Rotor Position Detection Method of SPMSM Based on New High Frequency Voltage Injection Method", IEEE Transactions on Power Electronics, 2019.

[78] Y. Q. Kang, R. D. Lorenz, "Using Intentional Magnetic Saturation for HFI Based Self-Sensing with SPMSMs", 2018 IEEE International Conference on Power Electronics, Drives and Energy Systems (PEDES), 2018.

[79] Xin Luo, Qipeng Tang, Anwen Shen, Qiao Zhang "PMSM Sensorless Control by Injecting HF Pulsating Carrier Signal Into Estimated Fixed-Frequency Rotating Reference Frame", IEEE Transactions on Industrial Electronics, 2016.

[80] W. Wang, Z. Li, X. Xu, "A novel smooth transition strategy for BEMF-based sensorless drive startup of PMSM" Proceeding of the 11th World Congress on Intelligent Control and Automation, IEEE, 2014.

[81] Dale Purves, George J Augustine, David Fitzpatrick, Lawrence C Katz, Anthony-Samuel LaMantia, James O McNamara, S Mark Williams, "Neuroscience", 2nd edition, Sunderland (MA), 2001.

[82] "Sine, Square, Triangle, Saw The Difference Between Waveforms and Why It Matters" https://www.perfectcircuit.com/signal/difference-between-waveforms accessed in 2022

[83]K. Kurakata, T. Mizunami, K. Matsushita, "Sensory unpleasantness of high-frequency sounds", Acoustical Science and Technology 34(1):26-33, January 2013.

[84]C. Nordqvist, "The Ten Noises We Hate The Most", https://www.medicalnewstoday.com/articles/251489 October 15, 2012

[85]Tomoharu Ishikawa, Muneto Tatsumoto, Katuhiro Maki, Minoru Mitsui, Hiroshi Hasegawa, Koichi Hirata2, "Identification of Everyday Sounds Perceived as Noise by Migraine Patients", National Institutes of Health, 2019.

[86]Hanchi Chen, "Theory and Design of Spatial Active Noise Control Systems", Ph.D Thesis, Australian National University, 2017.

[87]R. Hinchliffe, I. Scott, M. Purver, I. Stothers, “Tonal Active Control In Production on a Large Turbo-Prop Aircraft” INTER-NOISE and NOISE-CON Congress and Conference Proceedings 2002 (7), 369-376.

[88]S. Johansson, I. Claesson, S. Nordebo, P. Sjosten, "Evaluation of multiple reference active noise control algorithms on Dornier 328 aircraft data", IEEE Transactions on Speech and Audio Processing, vol: 7, issue:4, pp. 473 – 477, 1999.

[89]C. Howard, "Recent Developments in Submarine Vibration Isolation and Noise Control", Proceedings of 1st Submarine Science Technology and Engineering Conference, Adelaide, SA, Australia, 2011.

Glossary

γ ……………………….. Laplace correction constant

λ_m……………………….. Motor flux linkage

Ψ_f……………………….. Rotor flux

I_q……………………….. Motor current in the q axis of synchronous frame

I_d……………………….. Motor current in the d axis of synchronous frame

L_d……………………….. Motor inductance in the d axis of synchronous frame

L_q……………………….. Motor inductance in the q axis of synchronous frame

P ……………………….. Number of motor pole pairs

V_d……………………….. Motor voltage in the d axis of synchronous frame

V_q……………………….. Motor voltage in the q axis of synchronous frame

R_s……………………….. Motor stator resistance

List of figures

List of tables

Appendix A – Publications resulting from this research

The results from this research effort have been/to be presented in the publications listed below:

The proposed reliability enhancement strategy and the Pseudo Random High Frequency Injection (PRHFI) variant was presented in a paper at the IEEE WEMDCD21 conference:

A.1 "Hybrid Sensorless Motor Control in aerospace applications, a survey in sensorless control, optimizing for availability and acoustic noise", 5th IEEE Workshop on Electrical Machine Design, Control and Diagnosis, 8-9 APRIL 2021 WEMDCD21. The paper can be found in the IEEE xplore database.

The method of Active Noise Cancellation (ANC) by means of High Frequency Injection (HFI) has resulted into two publication submissions:

A.2 "Active Noise Control by means of High Frequency Injection in Electric Motors", journal paper submitted in 2023 to the IET Electric Power Applications. Review of the manuscript resulted into comments and the revised manuscript was re-submitted.

A.3 "A Novel Active Noise Cancellation Method for Electric Motors", was presented at the 12th International Conference on Power Electronics, Machines and Drives, IET PEMD23, 23-24 October 2023.

Appendix B – Code listings

B.1 Matlab source code

B.1.1 Resolver feedback simulation model

```
% Matlab m file to emulate resolver feedback signals as a function
% of the excitation input and rotor angle

clear;                   % reset variables
sin_cos_ampl = 5.0;      % definition of sin/cos amplitude

% Matlab sampling rate and simulation time
Fs = 100000;                   % samples per second
dt = 1/Fs;                     % seconds per sample
StopTime = 0.2;                % seconds
t1 = (0:dt:StopTime-dt)';      % Matlab time

% simulated motor electrical angle
theta=mod(pi*t1*30,2*pi);

% excitation signal frequency
excitation_signal_angle = (2*pi*1000*t1); % 1 kHz to enable visual inspection
of waveform

% resolver sin and cos feedback signals
res_cos_fb = sin_cos_ampl*cos(excitation_signal_angle).*cos(theta);
res_sine_fb = sin_cos_ampl*cos(excitation_signal_angle).*sin(theta);

% Plots
figure;
plot(t1,theta+43.0,'LineWidth',2)                          % rotor angle
hold on
plot(t1,res_cos_fb+37.2,'LineWidth',2);                    % cosine feedback
hold on
plot(t1,res_sine_fb+27.2,'LineWidth',2);                   % sine feedback
hold on
plot(t1,(sin_cos_ampl*cos(theta))+17.0,'LineWidth',2);     % cosine of rotor
angle, a number of ways exist to demodulate the signal at the controller
hold on
plot(t1,(sin_cos_ampl*sin(theta))+6.0,'LineWidth',2);    % sine of rotor angle,
a number of ways exist to demodulate the signal at the controller
hold off
set(gca,'XTick',[],'YTick',[]);
lgd = legend('Motor theta', 'Resolver cosine feedback', 'Resolver sine
feedback', 'Demod. cosine within controller', 'Demod. sine within controller');
lgd.FontSize = 12;
```

B.1.2 HEMAS BEMF voltage versus speed and angle simulation model

```
% This is a Matlab m file to illustrate in a graphical manner the
% mathematical relationship between rotor angle, rotor omega and back EMF
% voltages for the HEMAS motor

% setting up sampling rate, calculation iterations and simulation
% parameters
clear;
pole_pairs = 5;       % HEMAS pole pairs
ke=0.092;             % HEMAS Ke in V/rds
fs = 400000;
dt = 1/fs;
iterations = 1199999;         % calculation iterations
% iterations = 400000;        % zoomed in option

StopTime = iterations*dt;
t = (0:dt:StopTime)';
num=length(t);
R=zeros(num,1);
Q=zeros(num,1);
Z=zeros(num,1);
velo=zeros(num,1);
w=zeros(num,1);
n=zeros(num,1);
angle=zeros(num,1);
mech_angle=zeros(num,1);
angle_incr=0.0;            % initialization of emulated angle increment
acc_val = 0.0000000012;       % example acceleration value

% gradual increase of the rotor angle
for j = 1:iterations;
  angle(j+1)=angle(j)+angle_incr*pole_pairs;
  if (angle(j+1) > 2*pi)
    angle(j+1)=angle(j+1)-2*pi;
  end
  if (angle_incr < 0.0013613753381621432934811065538)
    angle_incr=angle_incr+acc_val;
  end
  mech_angle(j+1)=mech_angle(j)+angle_incr;
  if (mech_angle(j+1) > 2*pi)
    mech_angle(j+1)=mech_angle(j+1)-2*pi;
  end
end

% unwrapping of the angle and independent angle_incr calculation
Q=unwrap(mech_angle);
Z=radtodeg(Q);
velo=diff(Z)*fs;
w=velo/(360);
n=w*60;%rotations per minute
omega_calc=w*60;
omega_calc(num-1) = omega_calc(num-2);
omega_calc(num) = omega_calc(num-2);
```

```
% BEMF calculation as a function of ke, speed and rotor angle
Ealpha=-ke*(2*pi/60)*omega_calc.*sin(angle); % converting from rpm to rads
Ebeta=ke*(2*pi/60)*omega_calc.*cos(angle);   % converting from rpm to rads

% plotting
% Plot the signal versus time:
figure;

subplot(4,4,1:4);
plot(t,angle,'b', 'LineWidth',2) % scaling for plotting visibility
hold on
plot(t,mech_angle, 'color',[0.4660 0.6740 0.1880], 'LineWidth',2) % scaling for
plotting visibility
hold off
% lgd = legend('Motor electrical angle (rad)', 'Motor mechanical angle (rad)');
% lgd.FontSize = 13;

subplot(4,4,5:8);
plot(t, omega_calc, 'b', 'LineWidth',2) % scaling for plotting visibility
% lgd = legend('Motor speed (rpm)');
% lgd.FontSize = 13;

subplot(4,4,9:12);
plot(t, Ealpha, 'b', 'LineWidth',2) % scaling for plotting visibility
% lgd = legend('Back EMF voltage alpha axis (Volts)');
% lgd.FontSize = 13;

subplot(4,4,13:16);
plot(t, Ebeta, 'b', 'LineWidth',2) % scaling for plotting visibility
% lgd = legend('Back EMF voltage beta axis (Volts)');
% lgd.FontSize = 13;
```

B.1.3 Stationary Frame High Frequency Injection model

```
% This is a simple Matlab m file to illustrate in a graphical manner the
% the effect of stationary frame High Frequency Injection to the motor
% phase currents and motor torque

clear;
close;
Fs = 44100;                         % Sampling Frequency
dt = 1/Fs;                          % seconds per sample
t = (0:dt:0.05)';                   % seconds
inj_freq = 2000;                    % HFI frequency set to 2kHz
spin_freq_elec = 50;                % electrical frequency 50Hz
Ld = 15.0e-3;                       % d axis inductance
Lq = 37.0e-3;                       % q axis inductance
Pole_Pairs = 2;                     % No of Pole Pairs
Flux_link = 0.175;                  % Wb
Curr_ampl = 4;                      % Amplitude of 3 phase currents
producing spinning
HFI_curr_amplitude = 0.3;           % Amplitude of High Frequency Current
produced from HFI

% electrical angle
theta_elec = mod(2*pi*spin_freq_elec*t,2*pi);

% fundamental current waveforms
sinewave_spinning_a = Curr_ampl*sin(theta_elec+pi);
sinewave_spinning_b = Curr_ampl*sin(theta_elec+4*pi/3+pi);
sinewave_spinning_c = Curr_ampl*sin(theta_elec+2*pi/3+pi);

% HFI signals
sinewave_HFI_alpha = HFI_curr_amplitude*sin(2*pi*inj_freq*t);
sinewave_HFI_beta = 0.3*cos(2*pi*inj_freq*t);
sinewave_HFI_a = sinewave_HFI_alpha;
sinewave_HFI_b = -0.5*sinewave_HFI_alpha+0.5.*sqrt(3)*sinewave_HFI_beta;
sinewave_HFI_c = -0.5*sinewave_HFI_alpha-0.5.*sqrt(3)*sinewave_HFI_beta;
sinewave_HFI_d = sinewave_HFI_alpha .* cos(2*pi*spin_freq_elec*t) +
sinewave_HFI_beta .* sin(2*pi*spin_freq_elec*t);
sinewave_HFI_q = -sinewave_HFI_alpha .* sin(2*pi*spin_freq_elec*t) +
sinewave_HFI_beta .* cos(2*pi*spin_freq_elec*t);

% D Q axis current calculation producing spinning torque
Id = sinewave_spinning_a.*cos(theta_elec)+(sinewave_spinning_b-
sinewave_spinning_c).*sin(theta_elec)*1/sqrt(3);
Iq = -sinewave_spinning_a.*sin(theta_elec)+(sinewave_spinning_b-
sinewave_spinning_c).*cos(theta_elec)*1/sqrt(3);

% Torque calculation
Torque = (3/2) * Pole_Pairs * (Flux_link * (sinewave_HFI_q+Iq) + (Ld - Lq) *
(sinewave_HFI_d + Id) .* (sinewave_HFI_q+Iq));    % HF Torque is due to both d
and q axis current
HF_Torque = Torque-mean(Torque);

% plots
figure;
```

```
subplot(4,4,1:4);
plot(t,theta_elec,'LineWidth',2)
ax = gca;
ax.FontSize = 11;
title1 = title('Electrical angle');
title1.FontSize = 11;
ylim([0 2*pi])

subplot(4,4,5:8);
plot(t,sinewave_spinning_a,'LineWidth',2)
hold on
plot(t,sinewave_spinning_b,'LineWidth',2)
hold on
plot(t,sinewave_spinning_c,'LineWidth',2)
ax = gca;
ax.FontSize = 11;
title2 = title('Motor phase currents (no HFI)');
title2.FontSize = 11;
lgd = legend('Ia', 'Ib', 'Ic');
lgd.FontSize = 11;

subplot(4,4,9:12);
plot(t,sinewave_spinning_a+sinewave_HFI_a,'LineWidth',2)
ax = gca;
ax.FontSize = 11;
hold on
plot(t,sinewave_spinning_b+sinewave_HFI_b,'LineWidth',2)
ax = gca;
ax.FontSize = 11;
hold on
plot(t,sinewave_spinning_c+sinewave_HFI_c,'LineWidth',2)
ax = gca;
ax.FontSize = 11;
hold off
title3 = title('Motor phase currents (HFI stationary frame)');
title3.FontSize = 11;
lgd = legend('Ia\_HFI', 'Ib\_HFI', 'Ic\_HFI');
lgd.FontSize = 11;

subplot(4,4,13:16);
plot(t,HF_Torque ,'LineWidth',2)
hold on
ylim([-0.2 0.2]);
ax = gca;
ax.FontSize = 11;
title2 = title('High Frequency Torque');
title2.FontSize = 11;
```

B.1.4 Synchronous Frame High Frequency Injection model

```
% This is a simple Matlab m file to illustrate in a graphical manner the
% the effect of synchronous frame High Frequency Injection to the motor
% phase currents and motor torque

clear;
close;
Fs = 44100;                           % Sampling Frequency
dt = 1/Fs;                            % seconds per sample
t = (0:dt:0.05)';                     % seconds
inj_freq = 2000;                      % HFI frequency set to 2kHz
spin_freq_elec = 50;                  % electrical frequency 50Hz
Ld = 15.0e-3;                         % d axis inductance
Lq = 37.0e-3;                         % q axis inductance
Pole_Pairs = 2;                       % No of Pole Pairs
Flux_link = 0.175;                    % Wb
Curr_ampl = 4;                        % Amplitude of 3 phase currents
producing spinning
HFI_curr_amplitude = 0.3;             % Amplitude of High Frequency Current
produced from HFI

% electrical angle
theta_elec = mod(2*pi*spin_freq_elec*t,2*pi);

% fundamental current waveforms
sinewave_spinning_a = Curr_ampl*sin(theta_elec+pi);
sinewave_spinning_b = Curr_ampl*sin(theta_elec+4*pi/3+pi);
sinewave_spinning_c = Curr_ampl*sin(theta_elec+2*pi/3+pi);

% HFI signal
sinewave_HFI_d = HFI_curr_amplitude*sin(2*pi*inj_freq*t);
sinewave_HFI_q = 0;
sinewave_HFI_alpha = sinewave_HFI_d.*cos(2*pi*spin_freq_elec*t);
sinewave_HFI_beta = sinewave_HFI_d.*sin(2*pi*spin_freq_elec*t);
sinewave_HFI_a = sinewave_HFI_alpha;
sinewave_HFI_b = -0.5*sinewave_HFI_alpha+0.5.*sqrt(3)*sinewave_HFI_beta;
sinewave_HFI_c = -0.5*sinewave_HFI_alpha-0.5.*sqrt(3)*sinewave_HFI_beta;

% D Q axis current calculation producing spinning torque
Id = sinewave_spinning_a.*cos(theta_elec)+(sinewave_spinning_b-
sinewave_spinning_c).*sin(theta_elec)*1/sqrt(3);
Iq = -sinewave_spinning_a.*sin(theta_elec)+(sinewave_spinning_b-
sinewave_spinning_c).*cos(theta_elec)*1/sqrt(3);

% Torque calculation
Torque = (3/2) * Pole_Pairs * (Flux_link * (sinewave_HFI_q+Iq) + (Ld - Lq) *
(sinewave_HFI_d + Id) .* (sinewave_HFI_q+Iq));   % HF Torque is due to both d
and q axis current
HF_Torque = Torque-mean(Torque);

% plots
figure;
subplot(4,4,1:4);
plot(t,theta_elec,'LineWidth',2)
```

```
ax = gca;
ax.FontSize = 11;
title1 = title('Electrical angle');
title1.FontSize = 11;
ylim([0 2*pi])

subplot(4,4,5:8);
plot(t,sinewave_spinning_a,'LineWidth',2)
hold on
plot(t,sinewave_spinning_b,'LineWidth',2)
hold on
plot(t,sinewave_spinning_c,'LineWidth',2)
ax = gca;
ax.FontSize = 11;
title2 = title('Motor phase currents (no HFI)');
title2.FontSize = 11;
lgd = legend('Ia', 'Ib', 'Ic');
lgd.FontSize = 11;

subplot(4,4,9:12);
plot(t,sinewave_spinning_a+sinewave_HFI_a,'LineWidth',2)
ax = gca;
ax.FontSize = 11;
hold on
plot(t,sinewave_spinning_b+sinewave_HFI_b,'LineWidth',2)
ax = gca;
ax.FontSize = 11;
hold on
plot(t,sinewave_spinning_c+sinewave_HFI_c,'LineWidth',2)
ax = gca;
ax.FontSize = 11;
hold off
title3 = title('Motor phase currents (HFI Synchronous frame)');
title3.FontSize = 11;
lgd = legend('Ia\_HFI', 'Ib\_HFI', 'Ic\_HFI');
lgd.FontSize = 11;

subplot(4,4,13:16);
plot(t,HF_Torque ,'LineWidth',2)
hold on
ylim([-0.2 0.2]);
ax = gca;
ax.FontSize = 11;
title2 = title('High Frequency Torque');
title2.FontSize = 11;
```

B.1.5 HEMAS HFI demodulation model, simplified constant saliency

```
%
% Matlab model of sensorless position calculation using
% High Frequency sinusoidal injection to d axis RRF (assuming constant
saliency)
%
% injecting to d axis Vd_hf= -sin(ωc*t)
% resulting into
% i_q_est = Vcarrier*DL*sin(2*(theta_est-theta))*(cos(wc*t))/(ωc(ΣL^2-ΔL^2));
%
% ef = LPF(i_q_est*cos(wc*t))
%

clear;                                  % reset variables

theta_est = 0;                        % rotor's electrical angle
% theta_est = pi/3;                     % rotor's electrical angle
% theta_est = 2*pi/3;                     % rotor's electrical angle
% theta_est = 3*pi/3;                     % rotor's electrical angle
% theta_est = 4*pi/3;                     % rotor's electrical angle
% theta_est = 5*pi/3;                     % rotor's electrical angle

tolerance_angle_match = pi/1000; % tolerance to show in simulation that
Dtheta approaches 0

Ld = 1.069e-3;                          % inductance d axis
Lq = 1.158e-3;                          % inductance q axis
SL = (Ld + Lq) / 2;
DL = (Lq - Ld) / 2;

Vc_ov_fc = 50/1000; % Vcarrier*/wc

% generation of pseudo random or costant frequency HFI
iterations = 10000;                     % number of sinusoidal iterations to
simulate
lower_limit_frequency_prfi = 1000;    % lower limit of pseudo random frequency
injection
higher_limit_frequency_prfi = 1000;   % higher limit of pseudo random
frequency injection
fs = 48000;                             % Sampling frequency (samples per
second)
dt = 1/fs;                              % seconds per sample
StopTime = (1/lower_limit_frequency_prfi)*iterations; % seconds
t = (0:dt:StopTime)'; % seconds
num=length(t);
cos_HF=zeros(num,1);

rand_frequency=randi([lower_limit_frequency_prfi,higher_limit_frequency_prfi]
,iterations,1); % random frequency within +/-600Hz
last_start=1;

for j = 1:iterations;
  T = 1/(rand_frequency(j));
  actual_frequency = rand_frequency(j);
```

```
  for i=last_start:round(last_start+T/dt);
    cos_HF(i)=cos(2*pi*(rand_frequency(j))*t(i-last_start+1));
  end
  last_start=last_start+round(T/dt);
  if iterations < 100
    last_start_100_iterations = iterations;
  elseif j == 100
    last_start_100_iterations = last_start;
  end
end

cos_HF=cos_HF(1:last_start);           % removing zero elements in the end

t=t(1:last_start);                     % removing zero elements in the end

% emulating motor spinning
theta=mod(pi*t*0.5,2*pi);

% injected high frequency
i_q_est = Vc_ov_fc*DL*cos_HF.*sin(2*(theta-theta_est))/(SL^2 - DL^2);
i_q_hf_mult_cos = i_q_est.*cos_HF; % multiplying i_q_est with cos_wt

D = fdesign.lowpass('Fp,Fst,Ap,Ast',0.01,0.015,1,60);
Hd = design(D,'equiripple','StopbandShape','linear','StopbandDecay',20);
e_f = filter(Hd,i_q_hf_mult_cos);            % calculation of error

% indicate when theta_est is close to theta
angle_match = zeros(1,size(theta,2));
angle_match1 = zeros(1,size(theta,2));
for i = 1:length(theta)
angle_match(i)=theta_est;
    if (abs(theta(i)-theta_est) < tolerance_angle_match || abs(theta(i)-
theta_est) > 2*pi - tolerance_angle_match)
    angle_match1(i)=10.0;
  else
    angle_match1(i)=-10.0;
  end
end

% Plot the signal versus time:
figure;
subplot(4,4,1:8);
yline(0);
hold on
plot(t,theta)
hold on
plot(t,e_f*7)
hold on
plot(t,angle_match,'--','color',[0.9290 0.6940 0.1250],'LineWidth',1.5)
hold on
plot(t,angle_match1,'--','color',[0.9290 0.6940 0.1250],'LineWidth',1.5)
hold on

ax = gca;
ax.FontSize = 11;
```

```
ylim([-2*pi 2*pi])

subplot(4,4,9:12);
plot(t,i_q_est)
hold on
ax = gca;
ax.FontSize = 11;

subplot(4,4,13:16);
plot(t,i_q_hf_mult_cos)
ax = gca;
ax.FontSize = 11;

hold off
```

B.1.6 HEMAS HFI demodulation model, variable saturation modelling

```
%
% Matlab model of sensorless position calculation using
% High Frequency sinusoidal injection to d axis RRF (modelling varying
saturation)
%
% injecting to d axis Vd_hf= -sin(wc*t)
% resulting into
% i_q_est = Vcarrier*DL*sin(2*(theta_est-theta))*(cos(wc*t))/(ωc(ΣL^2-ΔL^2));
%
% ef = LPF(i_q_est*cos(wc*t))
%

clear;                          % reset variables

theta_est = 0;                      % rotor's electrical angle
% theta_est = pi/3;                   % rotor's electrical angle
% theta_est = 2*pi/3;                   % rotor's electrical angle
% theta_est = 3*pi/3;                   % rotor's electrical angle
% theta_est = 4*pi/3;                   % rotor's electrical angle
% theta_est = 5*pi/3;                   % rotor's electrical angle

tolerance_angle_match = pi/1000; % tolerance to show in simulation that angle
maytch is established

Ld_0 =  1.1930e-3; % d axis inductance point
Ld_1 =  1.1360e-3; % d axis inductance point
Ld_2 =  1.0690e-3; % d axis inductance point
Ld_3 =  1.0640e-3; % d axis inductance point

Lq_0 =  1.1940e-3; % q axis inductance point
Lq_1 =  1.1850e-3; % q axis inductance point
Lq_2 =  1.1580e-3; % q axis inductance point
Lq_3 =  1.1450e-3; % q axis inductance point

sat_thresh_0 =  0.000000000; % d axis current thresholds
sat_thresh_1 =  2.614672282; % d axis current thresholds
sat_thresh_2 =  5.209445330; % d axis current thresholds
sat_thresh_3 =  7.764571353; % d axis current thresholds

Id_HFI_satur = 5.21; % 5.21A is identified to be injected to estimated d axis

Ld = Ld_0;
Lq = Lq_0;

Vc_ov_fc = 50/1000; % Vcarrier*/wc

% generation of pseudo random or costant frequency HFI
iterations = 10000;                  % number of sinusoidal iterations to
simulate
lower_limit_frequency_prfi = 1000;   % lower limit of pseudo random frequency
injection
higher_limit_frequency_prfi = 1000;  % higher limit of pseudo random
frequency injection
```

```
fs = 48000;                          % Sampling frequency (samples per
second)
dt = 1/fs;                           % seconds per sample
StopTime = (1/lower_limit_frequency_prfi)*iterations; % seconds
t = (0:dt:StopTime)'; % seconds
num=length(t);
cos_HF=zeros(num,1);

DL = zeros(num,1);
SL = zeros(num,1);
Id = zeros(num,1);

voltage_HF_inj=zeros(num,1);
rand_frequency=randi([lower_limit_frequency_prfi,higher_limit_frequency_prfi]
,iterations,1); % random frequency within +/-600Hz
last_start=1;

for j = 1:iterations;
  T = 1/(rand_frequency(j));
  actual_frequency = rand_frequency(j);
  for i=last_start:round(last_start+T/dt);
    cos_HF(i)=cos(2*pi*(rand_frequency(j))*t(i-last_start+1));
    voltage_HF_inj(i)=rand_frequency(j)*sin(2*pi*(rand_frequency(j))*t(i-
last_start+1));
  end
  last_start=last_start+round(T/dt);
  if iterations < 100
    last_start_100_iterations = iterations;
  elseif j == 100
    last_start_100_iterations = last_start;
  end
end

cos_HF=cos_HF(1:last_start);          % removing zero elements in the end
voltage_HF_inj=voltage_HF_inj(1:last_start);

t=t(1:last_start);                    % removing zero elements in the end

% algorithm varrying estimated angle to observe error 0 aligns with
theta=mod(pi*t*0.5,2*pi);

for i = 1:num;
    Id(i) = cos(theta_est -
theta(i))*Id_HFI_satur*voltage_HF_inj(i)/lower_limit_frequency_prfi;
    if Id(i) <= sat_thresh_0
      Ld = Ld_0;
      Lq = Lq_0;
    elseif Id(i) > sat_thresh_0 && Id(i) <= sat_thresh_1
      Ld = Ld_0 + (Ld_1-Ld_0) * (Id(i) - sat_thresh_0)/(sat_thresh_1 -
sat_thresh_0);
      Lq = Lq_0 + (Lq_1-Lq_0) * (Id(i) - sat_thresh_0)/(sat_thresh_1 -
sat_thresh_0);
    elseif Id(i) > sat_thresh_1 && Id(i) <= sat_thresh_2
      Ld <= Ld_1 + (Ld_2-Ld_1) * (Id(i) - sat_thresh_1)/(sat_thresh_2 -
sat_thresh_1);
```

```
      Lq <= Lq_1 + (Lq_2-Lq_1) * (Id(i) - sat_thresh_1)/(sat_thresh_2 -
sat_thresh_1);
    elseif Id(i) > sat_thresh_2 && Id(i) <= sat_thresh_3
      Ld <= Ld_2 + (Ld_3-Ld_2) * (Id(i) - sat_thresh_2)/(sat_thresh_3 -
sat_thresh_2);
      Lq <= Lq_2 + (Lq_3-Lq_2) * (Id(i) - sat_thresh_2)/(sat_thresh_3 -
sat_thresh_2);
    end
    DL(i)=(Lq - Ld) / 2;
    SL(i)=(Ld + Lq) / 2;
end

% injected high frequency
t1=SL.*SL-DL.*DL;
i_q_est = Vc_ov_fc*DL.*cos_HF.*sin(2*(theta-theta_est))./t1;
i_q_hf_mult_cos = i_q_est.*cos_HF; % multiplying i_q_est with cos_wt

D = fdesign.lowpass('Fp,Fst,Ap,Ast',0.01,0.015,1,60);
Hd = design(D,'equiripple','StopbandShape','linear','StopbandDecay',20);
e_f = filter(Hd,i_q_hf_mult_cos);             % calculation of error

% indicate when theta_est is close to theta
angle_match = zeros(1,size(theta,2));
angle_match1 = zeros(1,size(theta,2));
for i = 1:length(theta)
angle_match(i)=theta_est;
  if (abs(theta(i)-theta_est) < tolerance_angle_match || abs(theta(i)-
theta_est) > 2*pi - tolerance_angle_match)
    angle_match1(i)=10.0;
  else
    angle_match1(i)=-10.0;
  end
end

% Plot the signal versus time:
figure;
subplot(4,4,1:8);
yline(0);
hold on
plot(t,theta)
hold on
plot(t,e_f*7)
hold on

plot(t,angle_match,'--','color',[0.9290 0.6940 0.1250],'LineWidth',1.5)
hold on
plot(t,angle_match1,'--','color',[0.9290 0.6940 0.1250],'LineWidth',1.5)
hold on
ax = gca;
ax.FontSize = 11;
ylim([-2*pi 2*pi])

subplot(4,4,9:12);
plot(t,i_q_est)
ax = gca;
```

```
ax.FontSize = 11;
hold on

subplot(4,4,13:16);
plot(t,i_q_hf_mult_cos)

ax = gca;
ax.FontSize = 11;

hold off
```

B.1.7 Infineon data processing and analysis script

Note that below is the Matlab script used to process the saliency hardware testing data from the Infineon DSP. The script was run six times to process the data for each of cases $\hat{\vartheta}$=0 rad, case2 $\hat{\vartheta}$= π/3 rad, case3 $\hat{\vartheta}$=2π/3 rad, case4 $\hat{\vartheta}$=π rad, case5 $\hat{\vartheta}$=4π/3 rad, case6 $\hat{\vartheta}$=5π/3 rad. To switch between the scripts line "theta_est = 0*pi/3" needs to be updated to support each of 6 testcases and equivalently the csv file name in the first case

```
clear;
theta_est = 0*pi/3; % estimated angle is 30 degrees
tolerance_angle_match = pi/100; % tolerance to show in simulation that Dtheta
approaches 0
fs=10e3;          % 10 kHz sampling
samples = 1000;   % 1000 samples
t = linspace(0, 0.1, samples);

filename = 'inj_1000Hz_sin_spinning20Hz_0pi_3.csv';

data = readtable(filename, 'NumHeaderLines',2);
data1=data(:,[2]);
table_data = table2array(data1);

Ia=table_data(1:samples);
Ib=table_data(1002:1002+samples-1);
Ic=table_data(2003:2003+samples-1);
theta=table_data(3004:3004+samples-1);
hw_err=table_data(4005:4005+samples-1);
hw_demod=table_data(6007:6007+samples-1);
hw_demod=hw_demod/20;

angle_match = zeros(1,size(theta,2));
angle_match1 = zeros(1,size(theta,2));
for i = 1:length(theta)
angle_match(i)=theta_est;
    if  (abs(theta(i)-theta_est)  <  tolerance_angle_match  ||  abs(theta(i)-
theta_est) > 2*pi - tolerance_angle_match)
    angle_match1(i)=40.0;
  else
    angle_match1(i)=-40.0;
  end
end

cos_HF = zeros(samples,1);
Iq_hp_mult_cos = zeros(samples,1);

invsqrt3=1/sqrt(3);
Id = Ia.*cos(theta)+(Ib-Ic).*sin(theta)*invsqrt3;
Iq = -Ia.*sin(theta)+(Ib-Ic).*cos(theta)*invsqrt3;
Iq_est = -Ia.*sin(theta_est)+(Ib-Ic).*cos(theta_est)*invsqrt3;

Iq_hp = highpass(Iq_est,800,fs);
Iq_hp_mult_cos = Iq_hp.*(hw_demod);
ef = lowpass(Iq_hp_mult_cos,100,fs);
```

```
figure;
subplot(4,4,1:8);
plot(t, theta);
hold on
plot(t, hw_err*0.08);
hold on
plot(t,angle_match,'--','color',[0.9290 0.6940 0.1250],'LineWidth',1.0)
hold on
plot(t,angle_match1,'--','color',[0.9290 0.6940 0.1250],'LineWidth',1.0)
hold on
ylim([-2*pi 2*pi])
ax = gca;
ax.FontSize = 7;

subplot(4,4,9:12);
plot(t, Id);
hold on
ylim([-7.0 7.0])
ax = gca;
ax.FontSize = 7;

subplot(4,4,13:16);
plot(t, Iq);
hold on
ylim([-10.0 10.0])
ax = gca;
ax.FontSize = 7;

hold off
```

B.1.8 Matlab model of the proposed PRHFI algorithm

This is a Matlab script illustrating the underlying Pseudo Random High Frequency Injection (PRHFI) algorithm proposed in chapter 7 of this thesis.

```
%
% Matlab model of Pseudo Random High Frequency Injection algorithm
%

clear;                          % reset variables

% generation of pseudo random or costant frequency HFI
iterations = 50;                 % number of sinusoidal iterations to simulate
lower_limit_frequency_prfi = 1000;    % lower limit of pseudo random frequency
injection
higher_limit_frequency_prfi = 3000;  % higher limit of pseudo random frequency
injection
max_DC_link = 270;
fs = 48000;                          % Sampling frequency (samples per second)
dt = 1/fs;                           % seconds per sample
StopTime = (1/lower_limit_frequency_prfi)*iterations; % seconds
t = (0:dt:StopTime)'; % seconds
t1 = (0:dt:StopTime)'; % seconds
num=length(t);

voltage_HF_inj=zeros(num,1);
voltage_HF_inj_lin=zeros(num,1);

rand_frequency=randi([lower_limit_frequency_prfi,higher_limit_frequency_prfi]
,2*iterations,1); % random frequency within +/-600Hz

last_start=1;

for j = 1:iterations;
  actual_frequency              =               lower_limit_frequency_prfi+(j-
1)*((higher_limit_frequency_prfi-lower_limit_frequency_prfi)/iterations);
  T = 1/actual_frequency;
  for i=last_start:round(last_start+T/dt);
    if actual_frequency < 2000

voltage_HF_inj_lin(i)=max_DC_link*(actual_frequency/higher_limit_frequency_pr
fi)*sin(2*pi*(actual_frequency)*t(i-last_start+1));
    else

voltage_HF_inj_lin(i)=0.7*max_DC_link*(actual_frequency/higher_limit_frequenc
y_prfi)*square(2*pi*(actual_frequency)*t(i-last_start+1));
    end
  end
  last_start=last_start+round(T/dt);
  if iterations < 100
    last_start_100_iterations = iterations;
  elseif j == 100
    last_start_100_iterations = last_start;
  end
end
```

```
voltage_HF_inj_lin=voltage_HF_inj_lin(1:last_start);
t1=t1(1:last_start);                          % removing zero elements in the end

last_start1=1;
for j = 1:2*iterations;
  if last_start1 >= last_start
      break
  end
  T = 1/(rand_frequency(j));
  actual_frequency = rand_frequency(j);
  for i=last_start1:round(last_start1+T/dt);
    if rand_frequency(j) < 2000

voltage_HF_inj(i)=max_DC_link*(rand_frequency(j)/higher_limit_frequency_prfi)
*sin(2*pi*(rand_frequency(j))*t(i-last_start1+1));
    else

voltage_HF_inj(i)=0.7*max_DC_link*(rand_frequency(j)/higher_limit_frequency_p
rfi)*square(2*pi*(rand_frequency(j))*t(i-last_start1+1));
    end
  end
  last_start1=last_start1+round(T/dt);
  if iterations < 100
    last_start_100_iterations = iterations;
  elseif j == 100
    last_start_100_iterations = last_start1;
  end
end

voltage_HF_inj=voltage_HF_inj(1:last_start);
t=t(1:last_start);                            % removing zero elements in the end

% Plot the signal versus time:
figure;
plot(t,voltage_HF_inj,'LineWidth',1.0)
ax = gca;
ax.FontSize = 7;
xlim([0 t(last_start)])

figure;
plot(t,voltage_HF_inj_lin,'LineWidth',1.0)
ax = gca;
ax.FontSize = 7;
xlim([0 t(last_start)])
```

B.1.9 Matlab audio analysis sinusoidal versus square wave injection

This is a Matlab script illustrating the acoustic difference between the noise from sinusoidal and square HFI.

```
%
% Matlab model illustrating acoustic differences between the
% acoustic noise from sinusoidal and square wave injection
%

clear
f_HFI = 1000;                   % 1000Hz HFI
fs = 48000;                     % Sampling frequency (samples per second)
dt = 1/fs;                      % seconds per sample
StopTime = 2.0;                 % seconds of audio sample
t = (0:dt:StopTime-dt)';        % seconds

sinusoidal_hfi_acoustics = sin(2*pi*f_HFI*t);          % Create waveform from
sinusoidal wave injection
square_hfi_acoustics = sawtooth(2*pi*f_HFI*t,1/2);    % Create waveform from
square wave injection

% plot the waveform
grid on

subplot(2,1,1);
plot(t,sinusoidal_hfi_acoustics)
xlim([0 0.005])
ax = gca;
ax.FontSize = 7;

subplot(2,1,2);
plot(t,square_hfi_acoustics)
xlim([0 0.005])
ax = gca;
ax.FontSize = 7;

sound(sinusoidal_hfi_acoustics, fs)
% sound(square_hfi_acoustics, fs)

audiowrite('1000Hz_sinewave_matlab.wav',sinusoidal_hfi_acoustics,fs);
audiowrite('1000Hz_squarewave_matlab.wav',square_hfi_acoustics,fs);
```

B.1.10 Matlab simulation of HFI ANC sliding frequency effect

This is a Matlab script simulating the resultant noise from two sinusoids of slightly different frequencies.

```
%
% Matlab code to illustrate the ANC effect of two sinusoids
% of slightly different frequencies
%
clear;
Fs = 48000;                         % Sampling Frequency
dt = 1/Fs;                          % seconds per sample
t = (0:dt:60)'; % seconds

% 1400Hz 10 sec period
inj_freq1 = 1399.9;
inj_freq2 = 1400;

% 1400Hz 5 sec period
% inj_freq1 = 1399.8;
% inj_freq2 = 1400;

% 1500Hz 10 sec period
% inj_freq1 = 1499.9;
% inj_freq2 = 1500;

% 1500Hz 5 sec period
% inj_freq1 = 1499.8;
% inj_freq2 = 1500;

% 1700Hz 10 sec period
% inj_freq1 = 1699.9;
% inj_freq2 = 1700;

% 1700Hz 5 sec period
% inj_freq1 = 1699.8;
% inj_freq2 = 1700;

% 2000Hz 10 sec period
% inj_freq1 = 1999.9;
% inj_freq2 = 2000;

% 2000Hz 5 sec period
% inj_freq1 = 1999.8;
% inj_freq2 = 2000;

sinewave_HFI_1 = 3.0*sin(2*pi*inj_freq1*t); % Create Tone
sinewave_HFI_2 = 3.0*sin(2*pi*inj_freq2*t); % Create Tone

figure;
subplot(3,3,1:3);
plot(t,sinewave_HFI_1 + sinewave_HFI_2)
ax = gca;
ax.FontSize = 10;
```

```
subplot(3,3,4:6);
plot(psd(spectrum.periodogram,sinewave_HFI_1,'Fs',Fs,'NFFT',length(sinewave_H
FI_1)));
ax = gca;
ax.FontSize = 10;
xlim([inj_freq1*0.001-0.001 inj_freq1*0.001+0.001])

subplot(3,3,7:9);
plot(psd(spectrum.periodogram,sinewave_HFI_2,'Fs',Fs,'NFFT',length(sinewave_H
FI_2)));
ax = gca;
ax.FontSize = 10;
xlim([inj_freq1*0.001-0.001 inj_freq1*0.001+0.001])
```

B.2 C source code for the HEMAS DSP platform

B.2.1 BEMF observer C code

Below is the C code for the Back EMF observer design that was run on the HEMAS DSP platform. Function `BEMF_demod_angle` is used to calculate the BEMF electrical angle and the code below the function was added within a separate function to calculate BEMF speed.

```
void BEMF_demod_angle(void) // skoul sensorless
{
  // calculate valpha_dem - e_alpha, vbeta_dem - e_beta
  // and scaling between voltage and current emulating motor R
  delta_alpha_scaled = (Valpha3_bemf_ref - est_BEMF_alpha);
  delta_beta_scaled = (Vbeta3_bemf_ref - est_BEMF_beta);

  // filter_value_LPF_alpha = k1_rl_sless*filter_value_LPF_alpha_last +
  // k2_rl_sless*delta_alpha_scaled;
  filter_value_LPF_alpha   =   k1_rl_sless_cfg*filter_value_LPF_alpha_last   +
k2_rl_sless_cfg*delta_alpha_scaled;
  filter_value_LPF_alpha_last = filter_value_LPF_alpha;
  filter_value_LPF_beta     =     k1_rl_sless*filter_value_LPF_beta_last     +
k2_rl_sless*delta_beta_scaled;
  filter_value_LPF_beta_last = filter_value_LPF_beta;
  est_BEMF_alpha = scaling_fact*filter_value_LPF_alpha - Ialpha3;
  est_BEMF_beta = scaling_fact *filter_value_LPF_beta - Ibeta3;
  // calculation of BEMF angle
  BEMF_angle_int = atan2f(-est_BEMF_alpha, est_BEMF_beta);
  BEMF_angle = BEMF_angle_int + M_PI + BEMF_offset;
  // from -pi to pi to 0 to 2pi
  if (om3_e_sl_f > 0) // BEMF_speed
  {
    BEMF_angle = BEMF_angle + M_PI;
    if (BEMF_angle > TWO_M_PI)
    {
      BEMF_angle = BEMF_angle-TWO_M_PI;
    }
  }
  else
  {
    BEMF_angle = BEMF_angle;
    if (BEMF_angle < 0)
    {
      BEMF_angle = BEMF_angle+TWO_M_PI;
    }
  }
}
```

B.2.2 Saliency observer C code

B.2.2.1 High Frequency Injection C source code

Below is the C code for the HFI that was run on the HEMAS DSP platform.

```
void PRHFI_gen(int low_freq, int high_freq, float amplitude_d) // skoul
sensorless
{
  if (low_freq == high_freq)
  {
    prhfi_freq = low_freq;
      theta_prhfi = theta_prhfi + TWO_M_PI*prhfi_freq*TS;
      while (theta_prhfi >= TWO_M_PI){
      if (prhfi_freq > 2000 || square_wav == 1)
      {
            theta_prhfi = 0.0;
      }
      else
      {
            theta_prhfi = theta_prhfi-TWO_M_PI;
        }
      }
      if (amplitude_d < 100)
      {
      prhfi_amplitude_scaled = amplitude_d;
      }
    else
    {
      prhfi_amplitude_scaled = 100;
    }
  }
  else                                              // Pseudo random HF
injection
  {
     theta_prhfi = theta_prhfi + 2*3.1459*prhfi_freq*TS;
       if (theta_prhfi >= 2*3.1459) // sinusoid generation complete change
frequency
     {
            if(PRHFI_cnt < (float)(prhfi_freq / low_freq)*2)
      {
            PRHFI_cnt = PRHFI_cnt + 1;
            if (prhfi_freq > 2000 || square_wav == 1)
            {
                theta_prhfi = 0.0;
            }
                  else
                  {
                    theta_prhfi = theta_prhfi-TWO_M_PI;
                  }
      }
      else
      {
            PRHFI_cnt = 0;
```

```
          // pseudo random generation
            Qt ^= Qt << 13;
            Qt ^= Qt >> 17;
            Qt ^= Qt << 5;
          prhfi_freq     =     (float)(low_freq)     +     (float)(high_freq-
low_freq)*(float)(Qt)/powf(2, 32); // scaling needed to ensure low_freq + Qt <
high_freq
            if (prhfi_freq > 2000 || square_wav == 1)
            {
                theta_prhfi = 0.0;
            }
                  else
                  {
                    theta_prhfi = theta_prhfi-TWO_M_PI;
                  }

        }
    }
      if (amplitude_d < 100)
      {
          prhfi_amplitude_scaled    =    amplitude_d    *    prhfi_freq    /
(float)(low_freq); // amplitude proportional to frequency
      }
    else
    {
        prhfi_amplitude_scaled = 100 * prhfi_freq / (float)(low_freq); //
amplitude proportional to frequency
    }
  }
  // square wave generation
  if (prhfi_freq > 2000 || square_wav == 1)
  {
        if (sinf(theta_prhfi) >= 0)
        {
            PRHFI_injection_d = -0.7 * prhfi_amplitude_scaled; // square -ve
voltage
          square_cnt = square_cnt + 1;
        }
        else
        {
          if (((int)((500000/prhfi_freq)) % 100) == 0) // injection half period
is a mutiple of PWM period
          {
                sq_pos_scaled = 0.7 * prhfi_amplitude_scaled;  // square +ve
voltage
          }
          else // apply amplitude compensation to positive square
          {
              sq_pos_scaled     =     0.7     *     prhfi_amplitude_scaled     *
((1/prhfi_freq)/2)/(((1/prhfi_freq)/2)-0.0001); // square +ve voltage amplitude
compensation
          }
            if (sq_pos_scaled > 100)
            {
             sq_pos_scaled = 100;
```

```
          }
              PRHFI_injection_d = sq_pos_scaled;
          square_cnt = 0;
        }
  }
  else
  {
      square_cnt = 0;
        PRHFI_injection_d  =  -prhfi_amplitude_scaled  *  sinf(theta_prhfi  +
ANC_offset);
  }
  theta_prhfi_fb = theta_prhfi + k_propagation_offset;
  theta_prhfi_fb = (theta_prhfi_fb >= 2*3.1459) ? (theta_prhfi_fb-2*3.1459) :
theta_prhfi_fb;

  if (prhfi_freq > 2000 || square_wav == 1)
  {
        if (cosf(theta_prhfi_fb) >= 0)
        {
              demod_sin = 0.7 * prhfi_amplitude_scaled; // k_propagation_offset
to be set during hardware testing
        }
        else
        {
              demod_sin   =   -0.7   *   prhfi_amplitude_scaled;   //
k_propagation_offset to be set during hardware testing
        }
  }
  else
  {
        demod_sin  =  prhfi_amplitude_scaled  *  cosf(theta_prhfi_fb);  //
k_propagation_offset to be set during hardware testing
  }
}



Valpha3_ref   =   Vd3_ref_dt*cosTheta3   -   Vq3_ref_dt*sinTheta3   +
PRHFI_injection_d*cosThetasens3;
Vbeta3_ref   =   Vd3_ref_dt*sinTheta3   +   Vq3_ref_dt*cosTheta3   +
PRHFI_injection_d*sinThetasens3 ;
```

B.2.2.2 Saliency observer demodulation C source code

Below is the C code for the saliency observer demodulation

```
void PRHFI_demod_angle(void) // skoul sensorless
{
  // High Pass filter to remove fundamental frequency = Iq_sless -
  // LPF(Iq_sless)
      Iq3_sensorless = Ibeta3*cosThetasens3 - Ialpha3*sinThetasens3;
    lpf_Iq3_sensorless = k1_prhfi_lpf_hpf_cfg*lpf_Iq3_sensorless_last +
                         k2_prhfi_lpf_hpf_cfg*Iq3_sensorless;
    lpf_Iq3_sensorless_last = lpf_Iq3_sensorless;
    hpf_Iq3_sensorless = Iq3_sensorless - lpf_Iq3_sensorless;

    i_q_hf_mult_cos = hpf_Iq3_sensorless * (demod_sin);
    // LPF to obtain angle error DC value by removing HF carrier from angle
    PRHFI_angle_error = k1_prhfi_err_cfg*PRHFI_angle_error_last +
                        k2_prhfi_err_cfg*i_q_hf_mult_cos;
    PRHFI_angle_error_last = PRHFI_angle_error;

    // PI control loop tracking the sensorless angle
    if (freeze_angle == 0)
    {
      thetasens3_e = theta3_e + thetasens3_err_cfg;
            PI_PRHFI_i_db = PI_PRHFI_i_db + Ki_PRHFI*PRHFI_angle_error*TS;
    }
      else if (freeze_angle == 1)
      {
            thetasens3_e = thetasens3_err_cfg;
            PI_PRHFI_i_db = 0;
      }
      else
      {
      PI_PRHFI_i_db = PI_PRHFI_i_db + Ki_PRHFI*PRHFI_angle_error*TS;
          thetasens3_e = thetasens3_e + (double) PI_PRHFI_i_db +
                         Kp_PRHFI*PRHFI_angle_error;
      }
      while (thetasens3_e >= 2*3.1459){
            thetasens3_e = thetasens3_e-2*3.1459;
      }
}
```

B.2.3 C code producing songs by means of HFI to the HEMAS platform

```
/* Frequency Notes in Hz
D_5 = 587, E_5 = 659, F#_5 = 740, G_5 = 784, A_5 = 880, B_5 = 988, D_6 = 1175,
C_6 = 1046 */
// Eine Kleine Octave 4, Frequency of Notes in Hz
const int freq_note2[119] = {392, 0, 294, 0, 392, 0, 294, 0, 392, 0, 294, 0,
392, 0, 494, 0, 587, 0, 523, 0, 440, 0, 523, 0, 440, 0, 523, 0, 440, 0, 370, 0,
440, 0, 294, 0, 392, 0, 392, 0, 494, 0, 440, 0, 392, 0, 392, 0, 370, 0, 370, 0,
440, 0, 523, 0, 370, 0, 440, 0, 392, 0, 392, 0, 494, 0, 440, 0, 392, 0, 392, 0,
370, 0, 370, 0, 440, 0, 523, 0, 370, 0, 392, 0, 392, 0, 392, 0, 370, 0, 330, 0,
370, 0, 392, 0, 392, 0, 494, 0, 440, 0, 392, 0, 440, 0, 494, 0, 494, 0, 587, 0,
523, 0, 494, 0, 523, 0, 587};
// Eine Kleine Octave 5, Frequency of Notes in Hz
const int freq_note3[119] = {784, 0, 587, 0, 784, 0, 587, 0, 784, 0, 587, 0,
784, 0, 988, 0, 1175, 0, 1046, 0, 880, 0, 1046, 0, 880, 0, 1046, 0, 880, 0,
740, 0, 880, 0, 587, 0, 784, 0, 784, 0, 988, 0, 880, 0, 784, 0, 784, 0, 740, 0,
740, 0, 880, 0, 1046, 0, 740, 0, 880, 0, 784, 0, 784, 0, 988, 0, 880, 0, 784,
0, 784, 0, 740, 0, 740, 0, 880, 0, 1046, 0, 740, 0, 784, 0, 784, 0, 784, 0,
740, 0, 659, 0, 740, 0, 784, 0, 784, 0, 988, 0, 880, 0, 784, 0, 880, 0, 988, 0,
988, 0, 1175, 0, 1046, 0, 988, 0, 1046, 0, 1175};
// Eine Kleine, Amplitude of HFI notes
const int amplitude_note2[119] = {50, 0, 50, 0, 50, 0, 50, 0, 50, 0, 50, 0, 50,
0, 50, 0, 50, 0, 50, 0, 50, 0, 50, 0, 50, 0, 50, 0, 50, 0, 50, 0, 50, 0, 50, 0,
50, 0, 50, 0, 50, 0, 50, 0, 50, 0, 50, 0, 50, 0, 50, 0, 50, 0, 50, 0, 50, 0,
50, 0, 50, 0, 50, 0, 50, 0, 50, 0, 50, 0, 50, 0, 50, 0, 50, 0, 50, 0, 50, 0,
50, 0, 50, 0, 50, 0, 50, 0, 50, 0, 50, 0, 50, 0, 50, 0, 50, 0, 50, 0, 50, 0,
50, 0, 50, 0, 50, 0, 50, 0, 50, 0, 50, 0, 50, 0, 50, 0, 50};
// Eine Kleine, Duration of HFI notes in units of x100 microseconds PWM period
const int note_duration2[119] = {7500, 10, 2500, 10, 7500, 10, 2500, 10, 2500,
10, 2500, 10, 2500, 10, 2500, 10, 10000, 10, 7500, 10, 2500, 10, 7500, 10, 2500,
10, 2500, 10, 2500, 10, 2500, 10, 2500, 10, 7500, 2000, 2500, 2500, 7500, 10,
2500, 10, 2500, 10, 2500, 10, 2500, 10, 2500, 10, 7500, 10, 2500, 10, 2500, 10,
2500, 10, 2500, 10, 2500, 10, 7500, 10, 2500, 10, 2500, 10, 2500, 10, 2500, 10,
2500, 10, 7500, 10, 2500, 10, 2500, 10, 2500, 10, 2500, 10, 2500, 10, 1300, 10,
1300, 10, 1300, 10, 2000, 10, 2500, 10, 2500, 10, 1300, 10, 1300, 10, 1300, 10,
1300, 10, 2500, 10, 2500, 10, 2000, 10, 2000, 10, 2500, 10, 2500, 10, 20000};

void play_audio(void)
{
  if  (enable_audio == 2)  // Song: Eine Kleine octave 4
  {
      if (note_index < 119)
      {
            if (delay_counter < note_duration2[note_index])
            {
                  delay_counter = delay_counter+1;
                  prhfi_low_freq_cfg = freq_note2[note_index];
                  prhfi_high_freq_cfg = freq_note2[note_index];
                  prhfi_amplitude_d_cfg = amplitude_note2[note_index];
            }
            else
            {
                  delay_counter = 0;
```

```
                note_index = note_index+1;
            }
        }
        else
        {
            delay_counter = 0;
            note_index = 0;
            prhfi_low_freq_cfg = 0;
            prhfi_high_freq_cfg = 0;
            prhfi_amplitude_d_cfg = 0;
        }
    }
    else if  (enable_audio == 3)  // Song: Eine Kleine octave 5
    {
        if (note_index < 119)
        {
            if (delay_counter < note_duration2[note_index])
            {
                delay_counter = delay_counter+1;
                prhfi_low_freq_cfg = freq_note3[note_index];
                prhfi_high_freq_cfg = freq_note3[note_index];
                prhfi_amplitude_d_cfg = amplitude_note2[note_index];
            }
            else
            {
                delay_counter = 0;
                note_index = note_index+1;
            }
        }
        else
        {
            delay_counter = 0;
            note_index = 0;
            prhfi_low_freq_cfg = 0;
            prhfi_high_freq_cfg = 0;
            prhfi_amplitude_d_cfg = 0;
        }
    }
    else if  (enable_audio == 4)  // reset HFI
    {
            delay_counter = 0;
            note_index = 0;
            prhfi_low_freq_cfg = 0;
            prhfi_high_freq_cfg = 0;
            prhfi_amplitude_d_cfg = 0;
    }
    else
    {
        delay_counter = 0;
        note_index = 0;
    }
}
```

B.3 VHDL source code for the HEMAS DSP platform

B.3.1 VHDL Linear Interpolation code for Ld, Lq

```
-- Ld, Lq motor saturation constants and signals
CONSTANT sat_thresh_0 : REAL :=  0.000000000;
CONSTANT sat_thresh_1 : REAL :=  2.614672282;
CONSTANT sat_thresh_2 : REAL :=  5.209445330;
CONSTANT sat_thresh_3 : REAL :=  7.764571353;
CONSTANT sat_thresh_4 : REAL :=  10.26060430;
CONSTANT sat_thresh_5 : REAL :=  12.67854785;
CONSTANT sat_thresh_6 : REAL :=  15.00000000;
CONSTANT sat_thresh_7 : REAL :=  17.20729309;
CONSTANT sat_thresh_8 : REAL :=  19.28362829;
CONSTANT sat_thresh_9 : REAL :=  21.21320344;
CONSTANT sat_thresh_10 : REAL := 22.98133329;
CONSTANT sat_thresh_11 : REAL := 24.57456133;
CONSTANT sat_thresh_12 : REAL := 25.98076211;
CONSTANT sat_thresh_13 : REAL := 27.18923361;
CONSTANT sat_thresh_14 : REAL := 28.19077862;
CONSTANT sat_thresh_15 : REAL := 28.97777479;

CONSTANT Ld_0 : REAL :=   1.1930e-3;
CONSTANT Ld_1 : REAL :=   1.1360e-3;
CONSTANT Ld_2 : REAL :=   1.0690e-3;
CONSTANT Ld_3 : REAL :=   1.0640e-3;
CONSTANT Ld_4 : REAL :=   1.0550e-3;
CONSTANT Ld_5 : REAL :=   1.0430e-3;
CONSTANT Ld_6 : REAL :=   1.0330e-3;
CONSTANT Ld_7 : REAL :=   1.0220e-3;
CONSTANT Ld_8 : REAL :=   1.0120e-3;
CONSTANT Ld_9 : REAL :=   1.0030e-3;
CONSTANT Ld_10 : REAL :=  0.9943e-3;
CONSTANT Ld_11 : REAL :=  0.9877e-3;
CONSTANT Ld_12 : REAL :=  0.9809e-3;
CONSTANT Ld_13 : REAL :=  0.9751e-3;
CONSTANT Ld_14 : REAL :=  0.9715e-3;
CONSTANT Ld_15 : REAL :=  0.9680e-3;

CONSTANT Lq_0 : REAL :=   1.1940e-3;
CONSTANT Lq_1 : REAL :=   1.1850e-3;
CONSTANT Lq_2 : REAL :=   1.1580e-3;
CONSTANT Lq_3 : REAL :=   1.1450e-3;
CONSTANT Lq_4 : REAL :=   1.1330e-3;
CONSTANT Lq_5 : REAL :=   1.1210e-3;
CONSTANT Lq_6 : REAL :=   1.1100e-3;
CONSTANT Lq_7 : REAL :=   1.0990e-3;
CONSTANT Lq_8 : REAL :=   1.0880e-3;
CONSTANT Lq_9 : REAL :=   1.0790e-3;
CONSTANT Lq_10 : REAL :=  1.0713e-3;
CONSTANT Lq_11 : REAL :=  1.0637e-3;
CONSTANT Lq_12 : REAL :=  1.0569e-3;
CONSTANT Lq_13 : REAL :=  1.0510e-3;
CONSTANT Lq_14 : REAL :=  1.0455e-3;
CONSTANT Lq_15 : REAL :=  1.0410e-3;
```

```
SIGNAL Ld : REAL := Ld_0; -- Motor phase inductance d axis
SIGNAL Lq : REAL := Lq_0; -- Motor phase inductance q axis


Ldq_calc: PROCESS(Id)
BEGIN
  IF Id <= sat_thresh_0 THEN
    Ld <= Ld_0;
    Lq <= Lq_0;
  ELSIF Id > sat_thresh_0 AND Id <= sat_thresh_1 THEN
    Ld <= Ld_0 + (Ld_1-Ld_0) * (Id - sat_thresh_0)/(sat_thresh_1 -
          sat_thresh_0);
    Lq <= Lq_0 + (Lq_1-Lq_0) * (Id - sat_thresh_0)/(sat_thresh_1 -
          sat_thresh_0);
  ELSIF Id > sat_thresh_1 AND Id <= sat_thresh_2 THEN
    Ld <= Ld_1 + (Ld_2-Ld_1) * (Id - sat_thresh_1)/(sat_thresh_2 -
          sat_thresh_1);
    Lq <= Lq_1 + (Lq_2-Lq_1) * (Id - sat_thresh_1)/(sat_thresh_2 -
          sat_thresh_1);
  ELSIF Id > sat_thresh_2 AND Id <= sat_thresh_3 THEN
    Ld <= Ld_2 + (Ld_3-Ld_2) * (Id - sat_thresh_2)/(sat_thresh_3 -
          sat_thresh_2);
    Lq <= Lq_2 + (Lq_3-Lq_2) * (Id - sat_thresh_2)/(sat_thresh_3 -
          sat_thresh_2);
  ELSIF Id > sat_thresh_3 AND Id <= sat_thresh_4 THEN
    Ld <= Ld_3 + (Ld_4-Ld_3) * (Id - sat_thresh_3)/(sat_thresh_4 -
          sat_thresh_3);
    Lq <= Lq_3 + (Lq_4-Lq_3) * (Id - sat_thresh_3)/(sat_thresh_4 -
          sat_thresh_3);
  ELSIF Id > sat_thresh_4 AND Id <= sat_thresh_5 THEN
    Ld <= Ld_4 + (Ld_5-Ld_4) * (Id - sat_thresh_4)/(sat_thresh_5 -
          sat_thresh_4);
    Lq <= Lq_4 + (Lq_5-Lq_4) * (Id - sat_thresh_4)/(sat_thresh_5 -
          sat_thresh_4);
  ELSIF Id > sat_thresh_5 AND Id <= sat_thresh_6 THEN
    Ld <= Ld_5 + (Ld_6-Ld_5) * (Id - sat_thresh_5)/(sat_thresh_6 -
          sat_thresh_5);
    Lq <= Lq_5 + (Lq_6-Lq_5) * (Id - sat_thresh_5)/(sat_thresh_6 -
          sat_thresh_5);
  ELSIF Id > sat_thresh_6 AND Id <= sat_thresh_7 THEN
    Ld <= Ld_6 + (Ld_7-Ld_6) * (Id - sat_thresh_6)/(sat_thresh_7 -
          sat_thresh_6);
    Lq <= Lq_6 + (Lq_7-Lq_6) * (Id - sat_thresh_6)/(sat_thresh_7 -
          sat_thresh_6);
  ELSIF Id > sat_thresh_7 AND Id <= sat_thresh_8 THEN
    Ld <= Ld_7 + (Ld_8-Ld_7) * (Id - sat_thresh_7)/(sat_thresh_8 -
          sat_thresh_7);
    Lq <= Lq_7 + (Lq_8-Lq_7) * (Id - sat_thresh_7)/(sat_thresh_8 -
          sat_thresh_7);
  ELSIF Id > sat_thresh_8 AND Id <= sat_thresh_9 THEN
    Ld <= Ld_8 + (Ld_9-Ld_8) * (Id - sat_thresh_8)/(sat_thresh_9 -
          sat_thresh_8);
    Lq <= Lq_8 + (Lq_9-Lq_8) * (Id - sat_thresh_8)/(sat_thresh_9 -
          sat_thresh_8);
```

```
  ELSIF Id > sat_thresh_9 AND Id <= sat_thresh_10 THEN
    Ld <= Ld_9 + (Ld_10-Ld_9) * (Id - sat_thresh_9)/(sat_thresh_10 -
          sat_thresh_9);
    Lq <= Lq_9 + (Lq_10-Lq_9) * (Id - sat_thresh_9)/(sat_thresh_10 -
          sat_thresh_9);
  ELSIF Id > sat_thresh_10 AND Id <= sat_thresh_11 THEN
    Ld <= Ld_10 + (Ld_11-Ld_10) * (Id - sat_thresh_10)/(sat_thresh_11 -
          sat_thresh_10);
    Lq <= Lq_10 + (Lq_11-Lq_10) * (Id - sat_thresh_10)/(sat_thresh_11 -
          sat_thresh_10);
  ELSIF Id > sat_thresh_11 AND Id <= sat_thresh_12 THEN
    Ld <= Ld_11 + (Ld_12-Ld_11) * (Id - sat_thresh_11)/(sat_thresh_12 -
          sat_thresh_11);
    Lq <= Lq_11 + (Lq_12-Lq_11) * (Id - sat_thresh_11)/(sat_thresh_12 -
          sat_thresh_11);
  ELSIF Id > sat_thresh_12 AND Id <= sat_thresh_13 THEN
    Ld <= Ld_12 + (Ld_13-Ld_12) * (Id - sat_thresh_12)/(sat_thresh_13 -
          sat_thresh_12);
    Lq <= Lq_12 + (Lq_13-Lq_12) * (Id - sat_thresh_12)/(sat_thresh_13 -
          sat_thresh_12);
  ELSIF Id > sat_thresh_13 AND Id <= sat_thresh_14 THEN
    Ld <= Ld_13 + (Ld_14-Ld_13) * (Id - sat_thresh_13)/(sat_thresh_14 -
          sat_thresh_13);
    Lq <= Lq_13 + (Lq_14-Lq_13) * (Id - sat_thresh_13)/(sat_thresh_14 -
          sat_thresh_13);
  ELSIF Id > sat_thresh_14 AND Id <= sat_thresh_15 THEN
    Ld <= Ld_14 + (Ld_15-Ld_14) * (Id - sat_thresh_14)/(sat_thresh_15 -
          sat_thresh_14);
    Lq <= Lq_14 + (Lq_15-Lq_14) * (Id - sat_thresh_14)/(sat_thresh_15 -
          sat_thresh_14);
  ELSE -- Id > sat_thresh_15
    Ld <= Ld_15;
    Lq <= Lq_15;
  END IF;
END PROCESS;

Ld_div_Lq <= Ld / Lq;
```

Appendix C – Transform equations

There are two widely used representations of a three-phase PMSM's currents and voltages, the stationary frame and the synchronous frame. The conversion between these representations are fundamental in motor control and are presented in this section of this document. The three-phase stationary frame representation is the voltages and currents at the stator's supply terminals $V_a, V_b, V_c, I_a, I_b, I_c$.

Clarke transform is a method used to convert from the three-phase stationary frame to the two-phase (alpha-beta) stationary frame as per below:

$$I\alpha = Ia \qquad \text{(C.1)}$$
$$I\beta = (Ib - Ic)/\sqrt{3} \qquad \text{(C.2)}$$
$$V\alpha = Va \qquad \text{(C.3)}$$
$$V\beta = (Vb - Vc)/\sqrt{3} \qquad \text{(C.4)}$$

Assuming the rotor's electrical angle is ϑ, Park transform is a method used to convert from the (alpha-beta) stationary frame to what is known as synchronous reference frame (d-q axis) as per below:

$$Id \ = \ I\alpha * cos\theta \ + \ I\beta * sin\theta \qquad \text{(C.5)}$$
$$Iq \ = \ -I\alpha * sin\theta \ + \ I\beta * cos\theta \qquad \text{(C.6)}$$
$$Vd \ = \ V\alpha * cos\theta \ + \ V\beta * sin\theta \qquad \text{(C.7)}$$
$$Vq \ = \ -V\alpha * sin\theta \ + \ V\beta * cos\theta \qquad \text{(C.8)}$$

In order to convert a motor's voltages and currents from the synchronous frame to the two phase (alpha-beta) stationary frame the Inverse Park transform is used:

$$V\alpha \ = \ Vd * cos\theta \ - \ Vq * sin\theta \qquad \text{(C.9)}$$
$$V\beta \ = \ Vd * sin\theta + \ Vq * \ cos\theta \qquad \text{(C.10)}$$
$$I\alpha \ = \ Id * cos\theta \ - \ Iq * sin\theta \qquad \text{(C.11)}$$
$$I\beta \ = \ Id * sin\theta + \ Iq * \ cos\theta \qquad \text{(C.12)}$$

Finally to convert a motor's voltages and currents from the two phase stationary frame to the three phase stationary frame the Inverse Clarke transform is used:

$$Va \ = \ V\alpha \qquad \text{(C.13)}$$
$$Vb = \ -(1/2)V\alpha \ + \ (\sqrt{3}/2)V\beta \qquad \text{(C.14)}$$
$$Vc = \ -(1/2)V\alpha \ - \ (\sqrt{3}/2)V\beta \qquad \text{(C.15)}$$
$$Ia \ = \ I\alpha \qquad \text{(C.16)}$$
$$Ib = \ -(1/2)I\alpha \ + \ (\sqrt{3}/2)I\beta \qquad \text{(C.17)}$$
$$Ic = \ -(1/2)I\alpha \ - \ (\sqrt{3}/2)I\beta \qquad \text{(C.18)}$$

Essentially, the three-phase stationary frame is a representation of voltages and currents in direct relation to the three physical power lines that connect the controller to the motor. The synchronous reference frame i.e. the d and q axis represent how the voltages and currents relate to the rotor's magnetic field. The synchronous frame is typically used when calculating the force exerted on the rotor named torque:

Some motor calculations are more clearly represented in one of the two frames. For example the torque exerted to the rotor for a PMSM is given by the below equation using phase currents in the synchronous frame.

Appendix D – Space Vector Modulation

Assuming a 3 phase PMSM and a 2 level inverter topology, 6 switches (switch_ph_a_up, switch_ph_a_low, switch_ph_b_up, switch_ph_b_low, switch_ph_c_up, switch_ph_c_low) are used to control the stator magnetic field and spinning of the motor. In order to align/couple the stator's magnetic field to the rotor's field, the stator field needs to be able to rotate around the full electrical angle range of 360^0. Activating each of the three phases at one time creates three vectors of magnetic flux at an angular interval of 120^0 each. A magnetic field of an arbitrary angle $0\text{-}360^0$ can be achieved by switching phases at a ratio defined by the required magnetic field angle. For example if a stator magnetic field is required that is half way (60^0) between phase A and phase B then phases B and C are switched on at a ratio of 50% and the voltage and flux vector observed by the motor is therefore at the required angle.

Going back to the 6 inverter switch topology, there is a total of 64 possible states of these switches. Excluding the illegal states that would results in a dangerous shoot through condition that shorts the DC link (switch_ph_N_up and switch_ph_N_low enabled simultaneously) there is a total of 6 active voltage vector combinations and 2 null vectors where "1" indicates that the top IGBT/Mosfet/SiCa for a phase is on and 0 the bottom listed below.

V1=001			V3=011			V2=010			V6=110			V4=100			V5=101		
1				1	1		1		1	1		1			1		1
a	b	c	a	b	c	a	b	c	a	b	c	a	b	c	a	b	c
	0	0	0			0		0			0		0	0		0	

Null vectors

V0=000			V7=111		
			1	1	1
a	b	c	a	b	c
0	0	0			

A sequence of V1, V3, V2, V6, V4, V5 generates voltages vector of 0, 60, 120,180,240,300 degrees vector. In order to obtain a voltage vector of intermediate angle the ratio of voltage vector varries as required to obtain a voltage vector of lower amplitude null vectors are used. Generally an svm elementary cycle forming a voltage vector's angle and amplitude is composed by a combination of 3 elements, two adjacent voltage vectors and a null vector. The proportion of each of these 3 elements constructs the angle and amplitude of the motor voltage.
Using v0 null vector can reduce the switching and switching losses as only two igbts are switching at one time v7 null vectors is not used as much as it has an igbt switched on continuously when running the motor at very slow speed what is used is alternated use of v0 and v7 null vectors.
The most widely used svm is the alternate reverse sequence which means follows a sequence of vx vy null null vy vx to reduce switching and losses and alternates using v0 and v7 null vectors so V1 V3 V7 V7 V3 V1 V0 V0 V1 V3 V7 V7 etc.

Appendix E – Audio Capturing Method

The acoustic capturing method used in chapters 6 and 7 of this thesis is based on a microphone included within a mobile phone (Samsung Galaxy J5). A number of microphone topologies and hardware architectures were trialed during this research and the embedded microphone within a phone device was found to be a good choice in terms of minimizing EMI from lab devices as no cables were used. The audio capturing was set to the sampling rate of 48kHz.

To evaluate the capabilities of the acoustic measurement method, this section illustrates capturing and processing results of audio at different frequencies. A set of audio files are therefore first generated using Matlab at frequencies of 1000Hz, 2000Hz, 3000Hz and played back using a laptop. Then the audio is captured using the method explained above and captured results are processed in Matlab.

The Matlab script that was used to generate the audio files is shown below

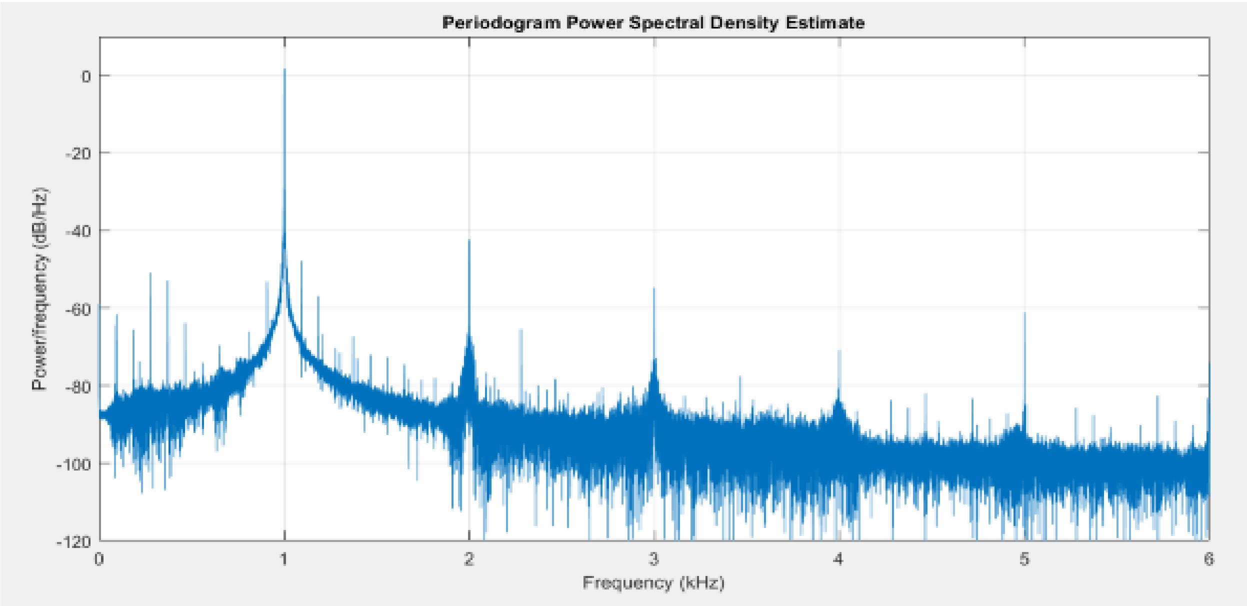
```
%
% Matlab code to save audio files of specified frequency
%
clear;
Fs = 48000;                          % Sampling Frequency
dt = 1/Fs;                           % seconds per sample
t = (0:dt:10)'; % seconds
sinusoid_freq = 1000.0;
% sinusoid_freq = 2000.0;
% sinusoid_freq = 3000.0;
sinewave = sin(2*pi* sinusoid_freq *t); % Create Tone
% Saving audio file of f_speaker to be played by laptop
audiowrite('500Hz_sinusoid.wav', sinusoid_freq,Fs);
```

The Matlab script that was used to process the FFT of the captured audio files is shown below:

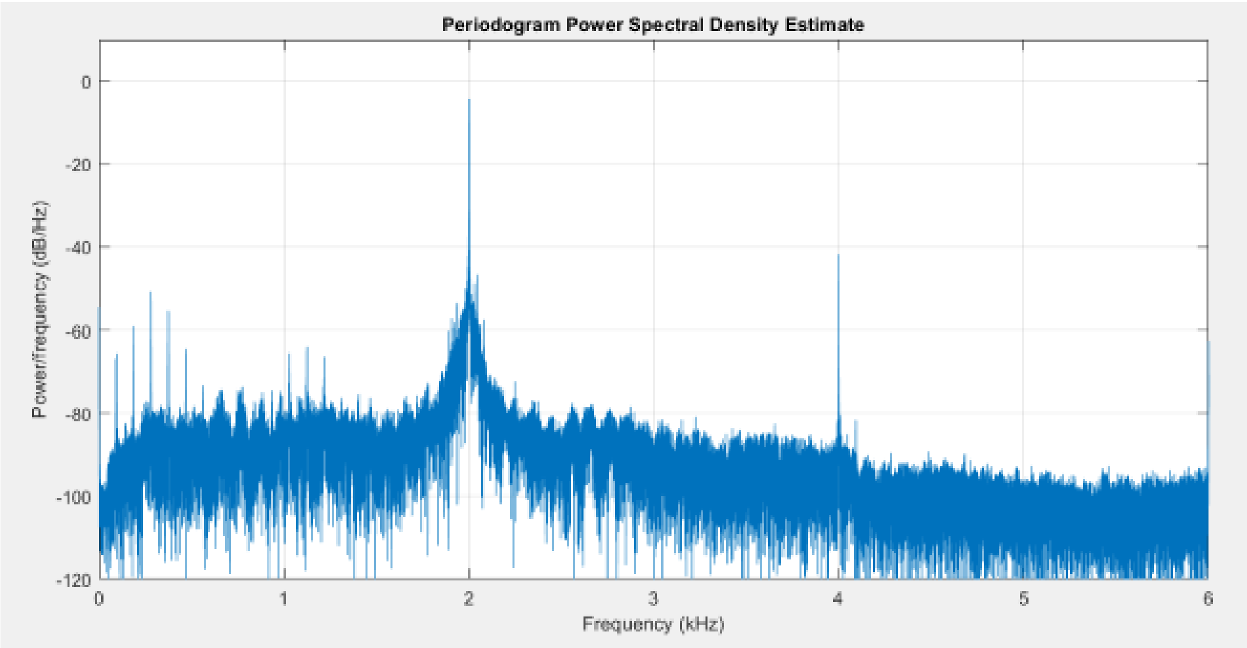
```
%
% Matlab script performing analysis to audio files
%
clear;

[y1,fs] = audioread('1khz.m4a');
% [y1,fs] = audioread('2khz.m4a');
% [y1,fs] = audioread('3khz.m4a');
figure;
samples = length(y1);
NFFT = 2^nextpow2(samples);         %// Next power of 2 from length of y
plot(psd(spectrum.periodogram,y1,'Fs',fs,'NFFT',length(y1)));
xlim([0 6.0])
ylim([-120 10])
ax = gca;
ax.FontSize = 10;
```

The screenshots below show the FFT of audio signals saved using the audio capturing method. One observation of the FFTs is that harmonics are observed along with the fundamental however they are relatively low amplitude specifically -40dB to -70dB. These harmonics are mainly due to the audio digitization process as well as sampling noise that results into the capturing of an imperfect sinusoid that appears as harmonics in the frequency domain.

1. Captured audio file of 1kHz waveform



2. Captured audio file of 2kHz waveform



3. Captured audio file of 3kHz waveform