Feasibility Study of

Virtually Distributed Hardware in The Loop

VCHV

Siamand Mostafavi

School of Engineering

Newcastle University

A thesis submitted for the degree of

Doctor of Philosophy

2024

Declaration

I, Siamand Mostafavi, hereby declare that the work presented in this thesis is genuine, done originally by me under the guidance and supervision of the acknowledged supervisors. To the best of my knowledge, I solemnly certify that I have not submitted this work elsewhere for the award of any degree under my name. I have generated the contents and materials of this thesis, along with the presented results, unless otherwise noted in the text, acknowledgements, and references. I certify that the content of this thesis, in the future, will not be used for submission under my name for any other degree or diploma in any university, except specified content that will be used for publication either in conferences or scientific journals.

December 2024

Abstract

Conventionally, a new vehicle powertrain assembly and subsystem integration took place once all the prototypes of different subsystems were designed and physically available. Despite accurate test results, the process was both time-consuming and costly. Particularly, the early prototypes usually required modification and, occasionally, redesign. Hardware in the loop introduces integration between hardware and models, allowing for early-stage integration. Electric drive systems commonly use hardware-in-the-loop simulation to simulate controllers before physically connecting them to the associated subsystem. By enabling integration over the internet or other communication mediums, distributed hardware in the loop elevates the methodology, enabling the integration of subsystems from geographically dispersed locations. This thesis does a feasibility study on a distributed hardware-in-the-loop simulation for an electric machine drive system. It tries out different configurations to see if spreading the subsystems out is possible. Also, the study investigates the impact of network impairment on electric drive systems.

Acknowledgements

Despite many obstacles, including personal matters, technical issues, financial difficulties, and the COVID-19 pandemic, I feel incredibly fortunate that I had wonderful people around me who supported me throughout my PhD. Words cannot express my gratitude to Dr. Matthew Armstrong, who unconditionally supported me on both a technical and personal level throughout my study. I would like to also express my gratitude to Prof. Barrie Mecrow, who has always been there to support me in difficult situations during these years as well as for helping me receive financial support. Undoubtedly, I would not be standing in this position without my supervisory team's invaluable guidance, patience, and feedback.

I also could not have undertaken this journey without IDE (Institute of Digital Engineering) and the members of the Virtually Connected Hybrid Vehicle Project, who generously provided knowledge and expertise. Additionally, this endeavour would not have been possible without their generous support and financing my research.

I am also grateful to my fellow PhD researchers, technicians in the UG lab, and the members of the mechanical workshop for their assistance in different stages of my project and moral support. In addition, I would like to show my utmost gratitude to Dr. Ari Al-Jaf for being extremely supportive and for proofreading this thesis.

Lastly, I would be remiss in not mentioning my family, especially my wife, my parents, and my siblings. Their belief in me has kept my spirits and motivation high during this process. I would also like to thank my sons for reminding me of the value of a restful night’s sleep and for giving me reasons to stay motivated during difficult situations.

Contents

Contents

Contents

List of Figures

List of Figures

CHAPTER 4

List of Figures

CHAPTER 5

CHAPTER 6

List of Figures

CHAPTER 7

CHAPTER 8

List of Figures

List of Tables

Chapter 1

Introduction

The automotive industry has experienced remarkable growth over the past few decades, as illustrated in Figure 1.1, which depicts the revenue of top automotive manufacturers worldwide, including Toyota [1], Volkswagen [2], General Motors [3], Ford [4], and Honda [5], during the first half of 2013 and the same period in 2023. The industry is projected to continue expanding, with revenues anticipated to reach approximately $4.8 trillion by 2033, driven by advancements in electric vehicle technology and a growing demand for premium vehicles [6]. According to [7], automation and electrification are the most significant changes occurring in the industry at present. As electric and autonomous vehicles become more prevalent, the global automotive market is becoming increasingly competitive, with new companies entering the market on a daily basis. To remain competitive in such a large industry, substantial investments in research and development, as well as a focus on improving new product development time scales, are necessary.

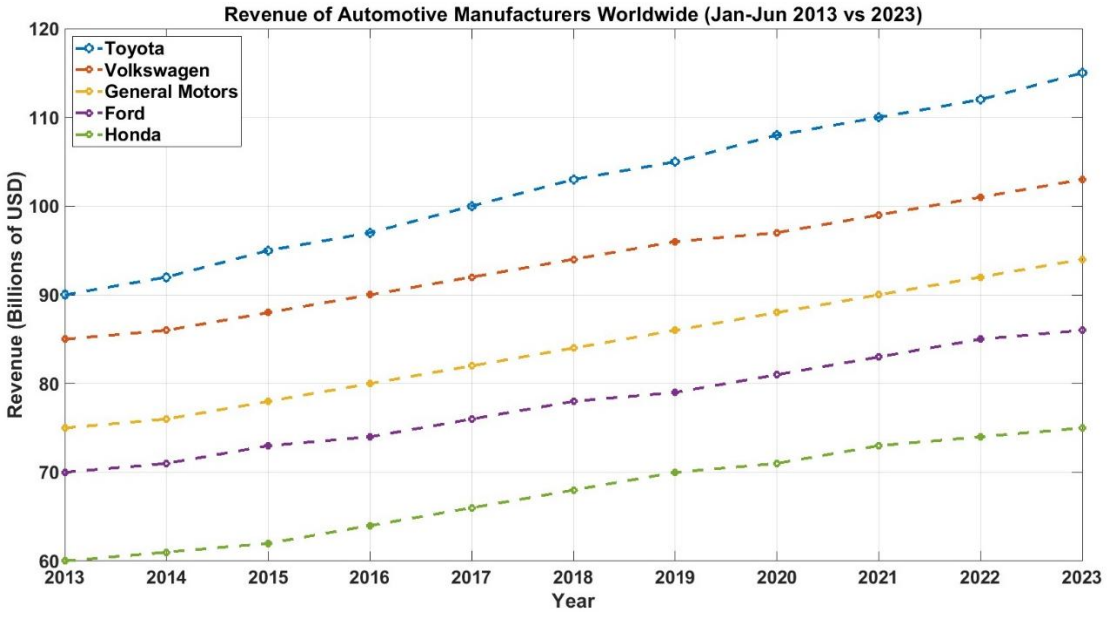

Figure 1. 1: Revenue of automotive top manufacturers worldwide between 2013 and 2023 [7]

On the other hand, global warming as a major global issue has compelled governments to pressurize companies within the automotive sector to examine fossil fuel alternatives. Since electricity generation has turned toward renewable energies and zero-emission nuclear energy, electrification appears to be the solution for zero- or lower-emission vehicles. [8] reveals that “current and future life-cycle emissions from electric vehicles and hydrogen-powered vehicles are on average lower than those of

new petrol cars and fossil boilers—not just on the global aggregate but also in most individual countries. Over time, in increasingly more regions, even the use of inefficient electric vehicles and hydrogen-powered vehicles is less emission-intensive than the most efficient new gasoline cars or fossil boilers".

The automotive industry appears destined for fundamental structural changes. Even though fully electric vehicles are the ultimate objective for the industry, hybrid vehicles emerge as a fundamental step toward their electrification. Hybrid vehicles are cheaper than electric vehicles due to their battery size; also, they benefit from having a longer range and less dependency on charging points, which was and still is one of the main concerns with electric vehicles [9], [10].

Furthermore, automotive giants are turning to modernization and are making changes to their approach toward new products to accelerate product development. Virtual engineering is an important factor in product development in the automotive industry. Not only does it allow for rapid prototyping, early-stage testing, and the integration of subsystems, but it also reduces cost and improves time consumption.

Traditionally, the assembly of a vehicle's physical powertrain occurs during the early stages of product development, approximately halfway through the completion of a new product. Although having a real prototype of the entire powertrain in the early stages can improve accuracy, the test results are often not ideal and necessitate further prototyping. Hence the traditional approach is time-consuming and wasteful.

Addressing this matter, virtual engineering is already attempting to postpone physical prototyping to later stages of vehicle development. As a result, virtual engineering brought various simulation and modeling techniques to the automotive industry.

Hardware in the loop (HiL) is a simulation method that is utilized to develop and validate a complex real-time embedded system. By accounting for the complexity of the plant under control, using a mathematical representation of all related dynamic systems, to the test platform, an effective platform will be formed. HiL provides the ability to work in a virtual environment where test conditions and vehicle behavior are simulated. Hence, by integrating the available physical subsystem with the virtual model of the inaccessible subsystem of the vehicle, one can test various control strategies without the actual vehicle. This method results in delaying physical prototyping significantly, which increases the chance of successful integration of the physical prototype of a vehicle's subsystems. [7] [11]

1.1. VCHV Project

The Virtually Connected Hybrid Vehicle (VCHV) is a four-year joint industry-academy project. The Digital Engineering and Test Centre (DETC) sponsors this unique collaborative UK national project, which aims to demonstrate the feasibility of real-time virtual hybrid powertrain testing using distributed Hardware in the Loop (HiL) simulation. The objective is to develop 6 different hybrid powertrain sub-systems (i.e., internal combustion engine, electric motor, battery, power electronic unit, and electronic unit controller), which will be distributed around the UK but virtually connected through the internet for real-time testing, as though they were an assembled power-train system.

As shown in Figure 1.2, eight PhDs from six of the UK's top universities—Bath, Loughborough, Newcastle, Nottingham, Warwick, and UCL—will deliver the HiL solution for their respective subsystems. [11]

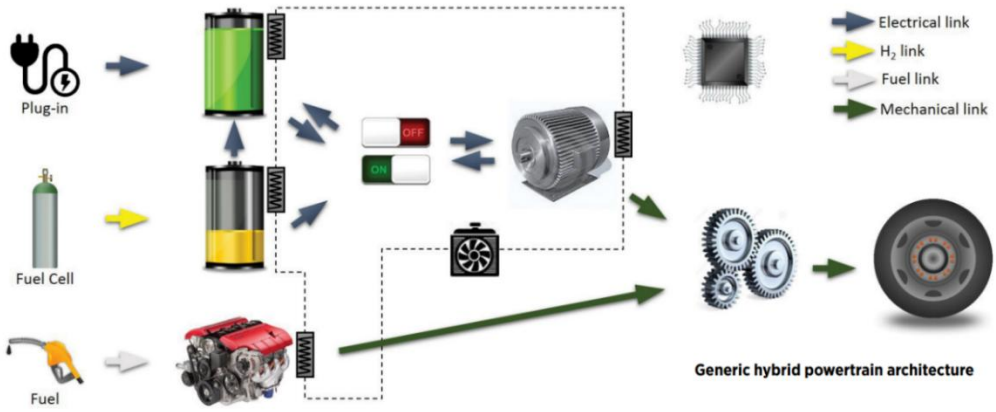

Figure 1. 2: VCHV - hybrid vehicle power train virtual connection [8]

- HV Controls and Communications - Loughborough University
- ICE - University of Bath
- Fuel Cells - University College London
- Electric Motors - University of Newcastle
- Power Electronics - University of Nottingham
- Batteries - University of Warwick

The project is aiming to contribute to further involvement of virtual engineering in the automotive industry and validate the feasibility of distributed hardware in the loop

simulation for automotive purposes. Distributed hardware-in-the-loop simulation allows for the integration of different subsystems in geographically dispersed locations through a reliable communication medium, specifically the Internet in this thesis case. Consequently, distributed hardware-in-the-loop simulation enhances time consumption by reducing the need to ship subsystems for integration, assemble prototypes, and make additional changes to various system components. This will be achieved by testing a virtual version of the complete system connected to different subsystems in different geographical locations via the Internet.

Furthermore, according to the initial proposal of the VCHV project, each facility develops a subsystem, also referred to as a unit under test (UUT), which must communicate in real time with other subsystems to form a complete powertrain. There will be two major stages to the project: local and global.

Local setup (Figure 1.3): Each facility will have a single real-time simulation processor (RTT: Real-Time Target) that is connected to the UUT and simulates hardware through an I/O module. The RTT will control and monitor the UUT through the I/O module. The emulator hardware will be used to emulate the upstream and downstream components in the global system; e.g., the power electronics converter (PEC) delivers the power generated from the battery to an electric motor: in this case, the PEC UUT would sit between a battery emulator and motor emulator.

[8] [9]

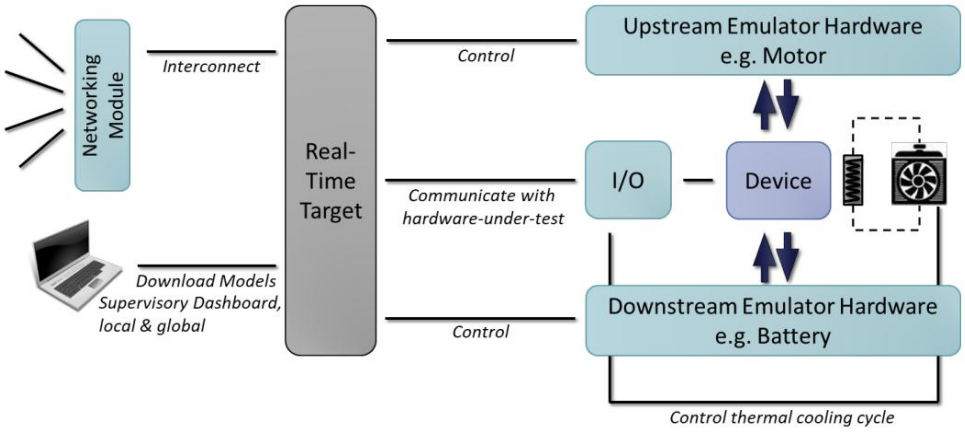

Figure 1. 3: Proposed local setup [8]

The global setup (Figure 1.4) assumes that the entire system consists of n sub-systems, with each RTT connected via LAN. The global setup will consist of two stages, as outlined below.

1. Software solution: The RTT at each location runs one master model (which follows the UUT behavior) and n-1 slave models (one for each remote RTT, simulating the corresponding UUT). The master model initially emulates the UUT.

2. The HiL solution involves replacing the master model with a UUT. At this stage, the UUT will receive signals from relevant subsystems via Remote Telemetry Transmission (RTT). There are challenges and concerns over data transfer and extraction that require consideration.

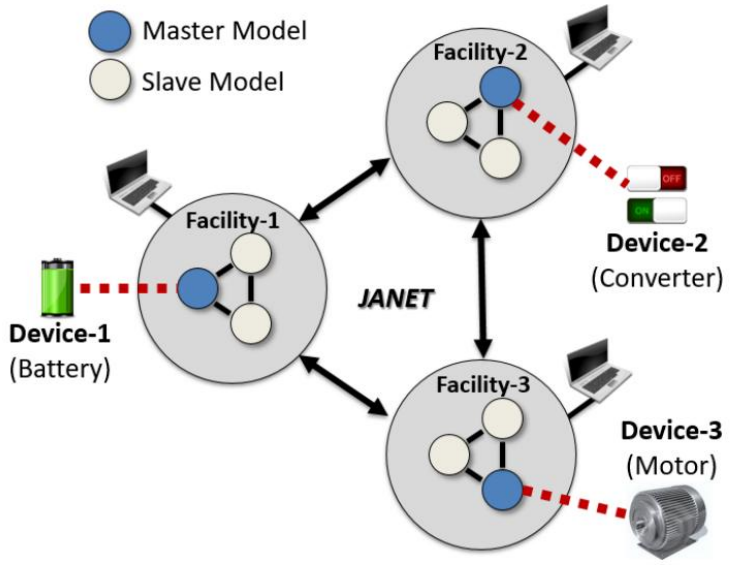

Figure 1. 4: Proposed global setup [8]

Initially, Newcastle University was particularly responsible for developing a Simulink model of an electric motor for the first step, as well as successfully utilizing a real-time simulation tool (predefined as RTT) that can be connected via the internet to the other components of the hybrid vehicle powertrain.

Even though the Simulink model and the real-time target remained in the final project proposal, collaboration with other VCHV members on a technical level for a distributed simulation was no longer possible for the following reasons. Firstly, the project had just launched in Newcastle, and the PhD student at Nottingham University was already in the final stage of his PhD. Additionally, it was believed that having a dynamic system like the electric machine in a distributed setup was ambitious and challenging at the time. Therefore, a more sophisticated and realistic proposal was needed to address the issue of integrating an electric machine into a distributed HiL setup. The following section explains and breaks down the proposal.

1.2. Overview of the refined Research Project Proposal

As previously stated, it was initially proposed to have an electric machine located in Newcastle University connected via the internet to associated power electronics and drive in Nottingham University. According to the previously discussed local and global setup, Newcastle University's setup would consist of an electric motor, Gate Drive, Inverter, and drive. The drive and power stage components are assigned as slaves, meaning that they are emulating a master model in Nottingham.

For the aforementioned motives in section1.1, a new proposal was inaugurated. The speed of a carrier signal of the electric machine drive system is in a region of 60µs while basic internet test reveals that despite using private IP addresses and utilizing JANET network that is a high-speed network for the UK research and education community. Latency can be as high as 8ms and 1ms jitter in communication between Newcastle University and Maidenhead, England. Latency is the time consumed by a packet to move from an endpoint to another, and jitter is the measure for the inconsistency of latency within a network. [10-12]

Therefore, implementing distributed hardware in the loop simulation on a dynamic system such as an electric machine drive system was a challenge to overcome in a more refined approach and could not be considered inadequately.

The new project proposal addresses distributed hardware in the loop in a more refined approach. The impression is to investigate the implementation of distributed Hardware in the loop, HiL, and software in the loop, Sil, simulation on an electric machine drive system. For this purpose, five main phases needed to be accomplished.

As mentioned previously, the electric machine that is used in this project is from a previous research project that addressed design of an SPM machine for an EPS system [13]. Figure 1.5 represent the acquired test rig and highlights different components.

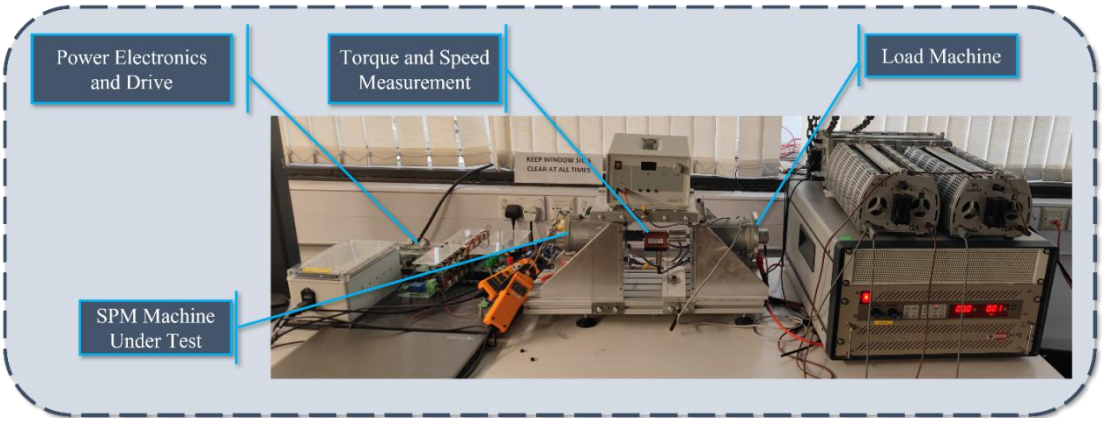

Figure 1. 5: Acquired test rig

The first step in the project was to develop an adequate virtual model of the test rig that can emulate the same performance as the hardware. Figure 1.6 represent the components of the virtual and hardware prototype. The test rig can be modelled at different fidelity to improve accuracy of the model. However, as fidelity of a model increases, the model gets more complex and requires more time for calculation as well as a more capable processor. Therefore, the model needed to be developed so that it can satisfy both accuracy requirement and matches the simulator capabilities.

Figure 1. 6: Hardware and virtual model components for the acquired test rig

The second phase of the project involves a real-time software-in-the-loop simulation, where a comprehensive model of the drive system is utilized to regulate an identical electric machine model (see Figure 1.7). In this process, the master current feedback is employed to determine the new current demand value calculated by the controller on the slave model. Furthermore, each model is loaded onto a distinct processing unit of the dSpace Scalexio simulator, and a 20-meter long Ethernet cable serves as the

communication link between the two setups. This section employs an approach to mitigate packet loss and latency in communication between the two models.

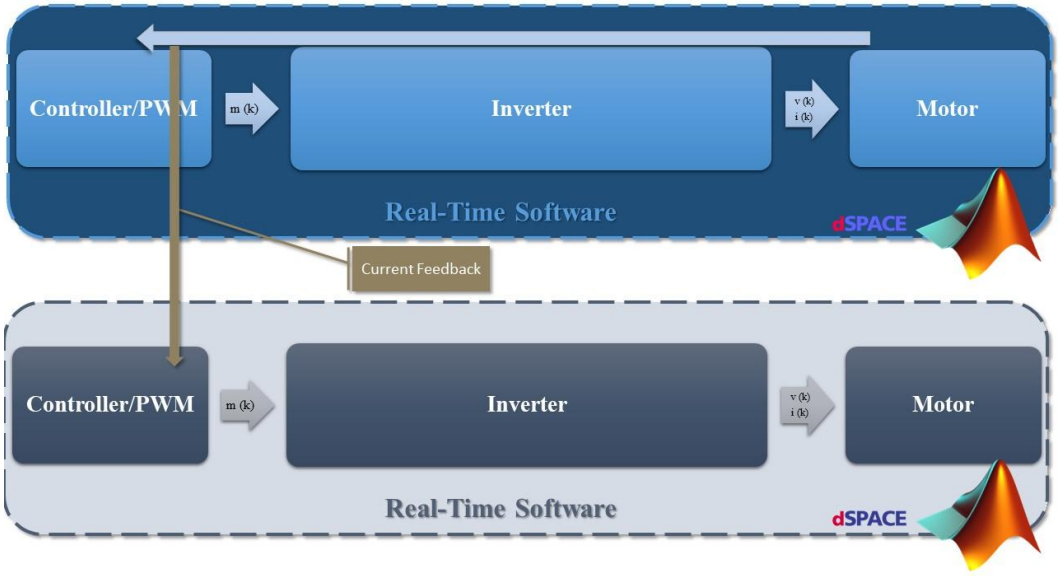

Figure 1. 7: The second phase of the project distributed real-time open-loop software-in-the-loop

After configuring the system, the next step is to add hardware to the setup in the form of a complete test rig, replacing the virtual master model. The current drawn by the SPM machine is fed into the dSpace scalexio and connected to the virtual model, where these values are used to operate the slave controller. This section is essential to validate the packet loss and latency solution. Figure 1.8 provides a visual representation of the setup.

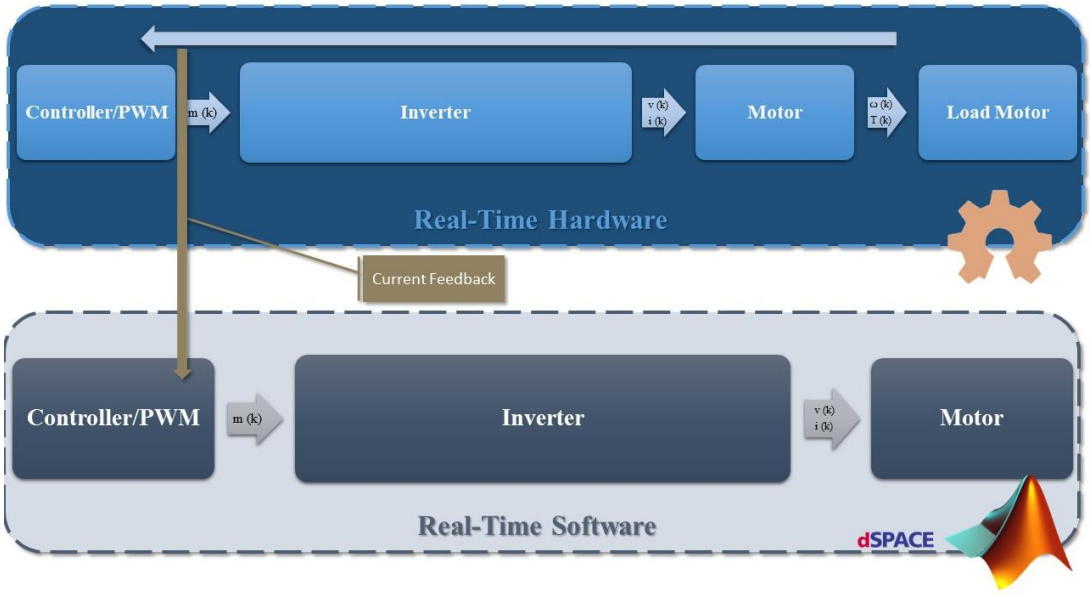

Figure 1. 8: The second phase of the project distributed real-time open-loop hardware-in-the-loop

During the third phase of the project, the slave model controller and power electronic models were eliminated, and a three-phase voltage signal was directly fed from the master model into the slave model, which incorporated the SPM machine (as shown in Figure 1.9). This setup necessitated a novel strategy and extensive calibration, as the voltage signal's frequency was considerably greater than the current feedback signal in the previous section.

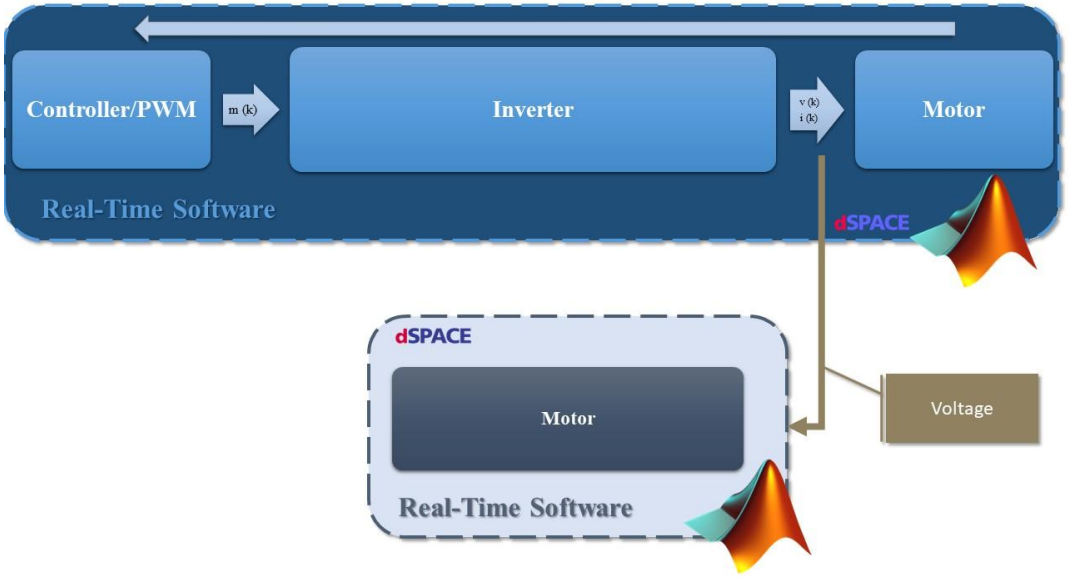

Figure 1. 9: The fourth phase of the project, distributed real-time open-loop software-in-the-loop.

Lastly, the configuration was established to enable a real-time, distributed closed-loop simulation, shown in Figure 1.10. In this simulation, not only does the voltage signal control the electric motor slave model, but there is also a communication node between the slave and master models that sends the torque feedback signal from the slave model to the master model controller, thus forming a closed-loop real-time software-in-the-loop simulation. This section provides an opportunity to validate the developed communication architecture in a closed-loop system and assess its robustness.

Figure 1. 10: Final Stage of The Project, Distributed Real-Time Hardware-In-The-Loop

Distributed real-time HiL simulation concept for EPS electric motor drive system can be crucial for testing different components of an EPS system inside a vehicle without the necessity to integrate the component into the overall system. Also, it allows for testing a component in the system when there is still some component physically missing. The virtual environment allows for product evaluation at a minimised cost and time consuming, also with reasonably high accuracy and fidelity.

Furthermore, a distributed version of HiL will include all the aforementioned benefits of HiL, along with additional advantages such as further cost and time reduction, as well as facilitating quick prototyping and integration throughout the product development stage. The feasibility analysis conducted in this thesis is essential not just for electric drive systems but also for other complex systems that need advanced communication to address simulation challenges like packet loss, delay, and jitter. Furthermore, the research may be reproduced for highly dynamic systems.

1.3. Overview of thesis:

This thesis includes seven chapters, which start with chapters one and two, which include a review of preceding research and fundamental knowledge related to the research project. Chapter three explains the setup employed in the project and the basics of the real-time simulator. Chapter four explains the Simulink model developed for this study and validates it by comparing the results with the actual hardware measurement.

Moreover, Chapter Five discusses the impact on the electric drive system when network issues arise during communication. Chapter six represents the novel methodology that undertakes resolution to network problems, including packet loss, jitter, and latency. In chapter seven, the research project is concluded by summarizing its challenges and exploring potential future opportunities.

1.4. Publications:

The following paper was submitted and presented in the 11th International Conference on Power Electronics, Machines and Drives (PEMD 2022):

1. S. Mostafavi and M. Armstrong, "Distributed hardware in the loop simulation for electric drive system," 11th International Conference on Power Electronics, Machines and Drives (PEMD 2022), Hybrid Conference, Newcastle, UK, 2022, pp. 102-106, doi: 10.1049/icp.2022.1025.

Chapter 2

Preceding Research

Evolving automotive industry is moving toward electrification that has increased the complexity of systems within a vehicle. Developing a new vehicle requires precise design and investment. Considering the high number of components in a vehicle, the process of integrating every component to form different subsystems and integrating major subsystems to form the final successful product is challenging. Integration in automotive product development is costly and time-consuming, which considering the current state of competition in the industry is an important factor. [14]

Conventionally, different systems and their associated subsystem were designed and manufactured individually then integration of the systems were taking place in the final stages of a vehicle's development, Figure 2.1. despite precise accuracy of testing in different stages, different systems required minor or major redesigning due to failure in integration stages. Thus, this method was time consuming, costly, and hazardous.

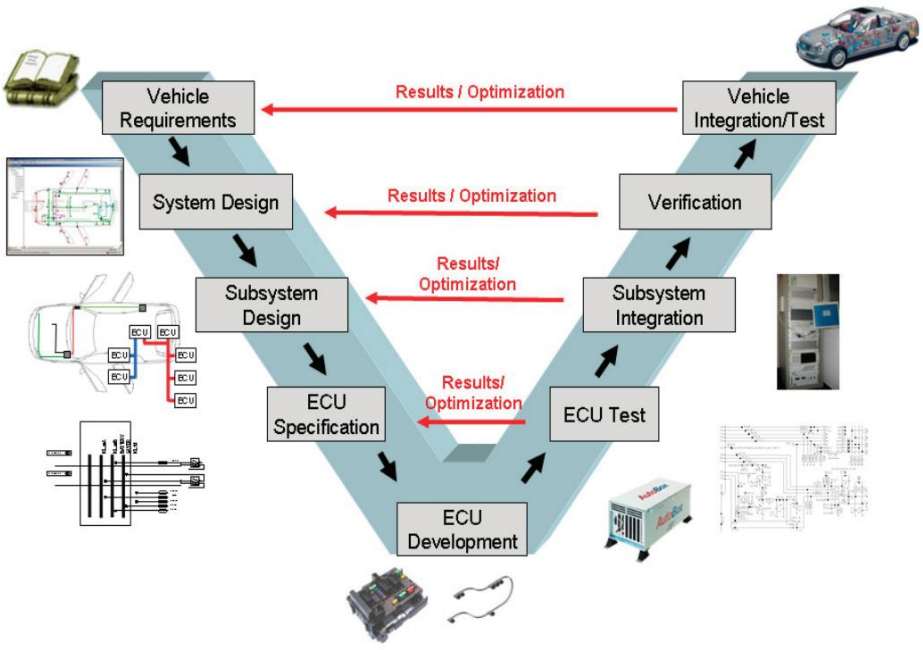

Figure 2. 1: Conventional vehicle development [14]

In the current product development procedure, new techniques are used to improve product development time, cost-effectiveness and safety. Hardware in the loop (HiL) simulation is a technique where a simulator is used to test a piece or pieces of a system that are software base models with physical hardware. [14]

Hardware in the loop has been around for decades and it has been used in different application areas that include automotive [15, 16] aerospace [17, 18], manufacturing [19], robotics[20] and army technology [21]. Many key features make HiL simulation such an attractive simulation technique. replacing some hardware of a system with software-based models significantly reduces the cost required for purchasing, transferring, and integrating the whole system. Another element that complies with the cost-effectiveness of HiL simulation is the non-destructive nature of HiL simulation, which also increases the safety of the simulation. Also, real-time HiL simulation allows for rapid prototyping and enables repeatability of simulation.[15]

Furthermore, HiL has proven to be reliable and capable of handling high fidelity simulation. Moreover, HiL allows for a wide variety of scenarios and testing conditions, as well as allowing simultaneous system engineering that enables different teams to develop different parts of the system while having the system integrated throughout all stages of development. [15]

To achieve a successful high-fidelity hardware-in-the-loop simulation, there are certain requirements to consider, Figure 2.2. According to [15], eight elements play key roles in hardware in the loop simulation.

- Actuator fidelity and sensor, unobtrusiveness is key for successful HiL simulation. Sensors and actuators are significant for a fast and precise measurement while unobtrusiveness accounts for resistance of the underlying hardware to change of dynamics.
- Signal conditioning and digital signal processing usually take place on the HiL simulator interface boards.
- Fixed step integration, computationally fast and capable processor and real-time operating system are other requirements for an effective HiL simulation.
- Sophisticated modelling plays a major role in a successful HiL simulation.
- Advanced laboratory instrument and diagnostic systems
- Virtual models' stiffness.
- Communication networks are required to be high in bandwidth to enable fast communication, especially in more dynamic systems.
- Synergism between hardware prototype and virtual models need to be achieved with the simulator for a successful HiL simulation.

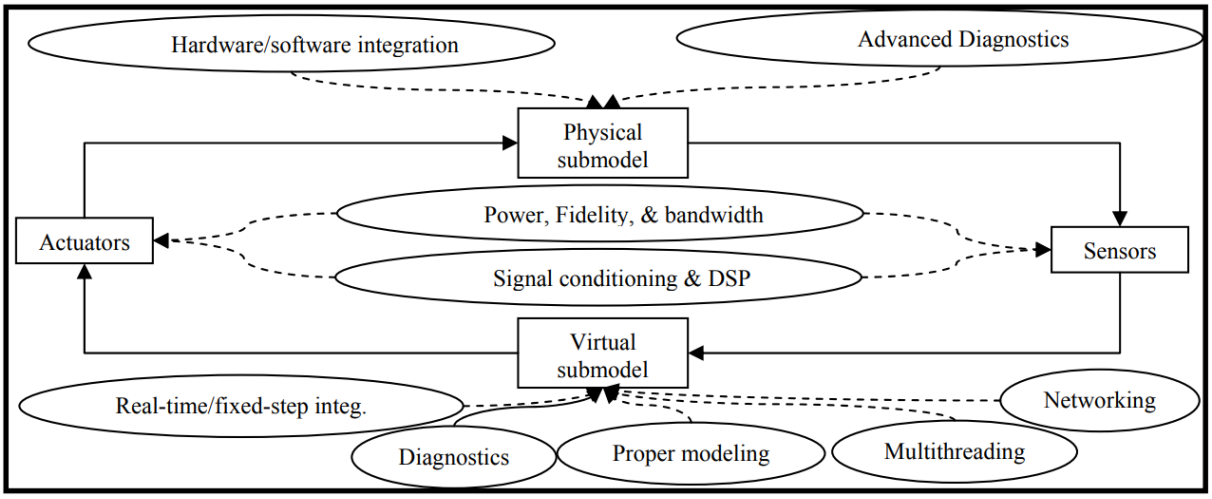

Figure 2. 2: HiL simulation enabler [15]

HiL simulation has become a vital part of the vehicle development process. During the past decades since the early time of HiL simulation 1950s, the hardware in the loop technique has developed with the development of computer processing units, and different companies such as dSpace, National Instrument, Triphase, Typhoon HIL, etc. have been able to manufacture highly capable real-time HiL simulators. With the new generations of the Internet becoming more reliable as a fast communication medium, having geographically distributed HiL simulation is becoming an attractive and accessible ambition. [22]

“Networking different HIL simulators will allow different engineering teams to design selected subsystems independently and then integrate their design efforts through the Internet, thereby demonstrating global concurrent systems engineering. Suppliers, OEMs and customers will have an opportunity for much more reliable evaluations of designs and concepts early in the development stage, thus facilitating strategic decisions and partnering.” [23]

In 2009, Ersal et al. [15] successfully integrated two setups that include an engine-in-the-loop at the University of Michigan in Ann Arbor and driver in the loop ride motion simulator at the US At Army Tank-Automotive Research, Development and Engineering Centre (TARDEC) in Warren, MI, The USA. During this project, the two setups were successfully integrated into an observer-free, event-based real-time simulation, Figure.2.3 is a visual representation of the HiL models.

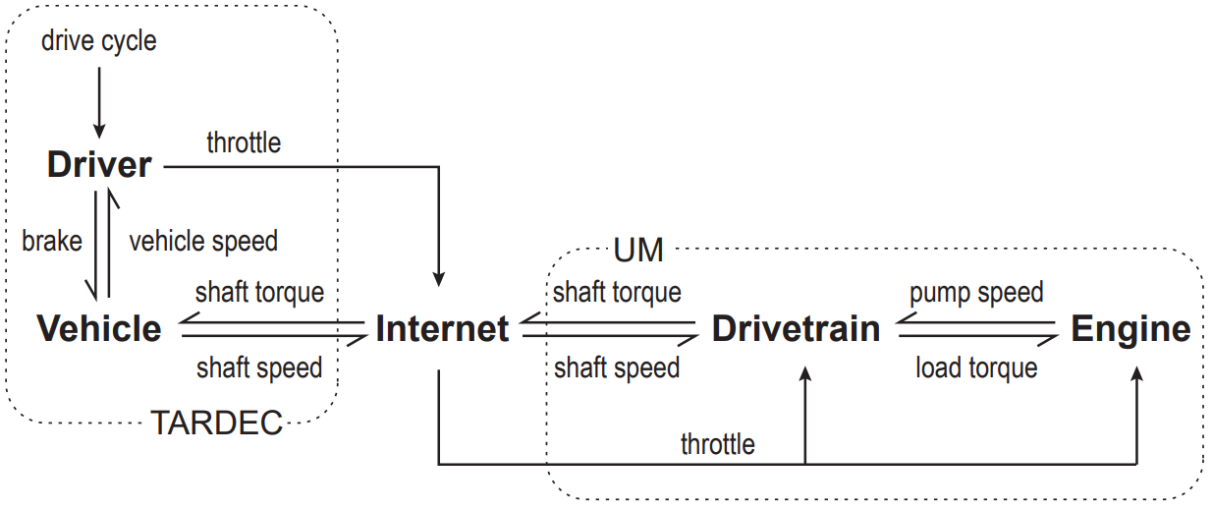

Figure 2. 3: Overview of UM-TARDEC *HiL* setup model [15]

Despite the low-frequency signal, 20Hz, that was transferred between the two setups, this research ignited further interest in deploying Internet Distributed HiL simulation, ID-HiL, for automotive applications. Inspired by the work underwent in [21] and [23], this thesis is exploiting the feasibility of having distributed HiL simulation for an electric machine drive system. The thesis focuses on communication protocol/architecture, model fidelity, different coupling points as well as packet prediction.

2.1 Communication Protocol/Architecture

In [24], Yi Zhang explained how he implemented the concept of data coupling on hardware in the loop simulation for a hybrid vehicle. The research focuses on internet distributed HiL simulation, where there are five subsystems. Figure 2.4 depicts the setup in [24] and highlights the subsystems; vehicle in the loop, driver in the loop, engine in the loop, battery in the loop and the cloud. The subsystems with the italic label in Figure 2.4 are software base, electric machine, and drivetrain, while the rest are physically present, driver, vehicle, engine management system, engine and battery.

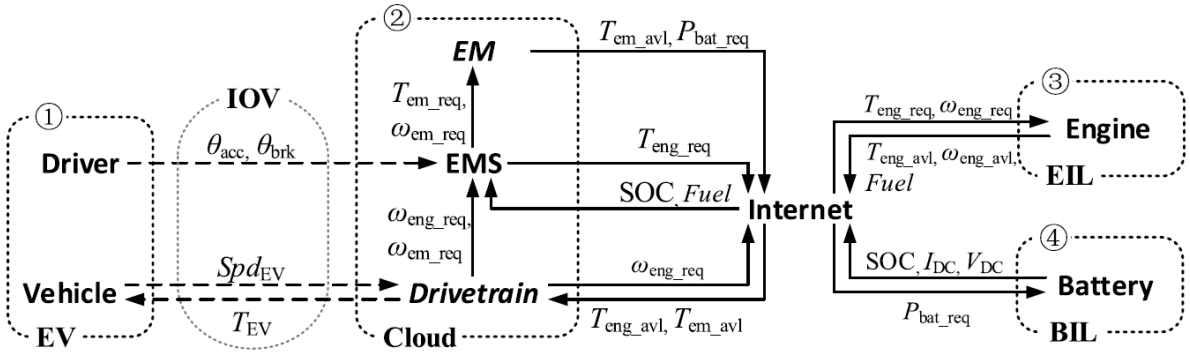

Figure 2. 4: Internet *HiL* setup in [24]

Communication analysis in [24] exploits both user datagram protocol (UDP) and transmission control protocol (TCP). The UDP protocol has the benefit of minimum delay on its side, on the other hand, TCP is more reliable as each packet is guaranteed delivery. Zhang claims that there is a negligible increase in transmission delay between UDP and TCP that is measured to be 5.8ms, Figure 2.5. Even though for a slow system such as the ones used in [24] the measured transmission delay is negligible, for an electric machine drive system this delay is vital.

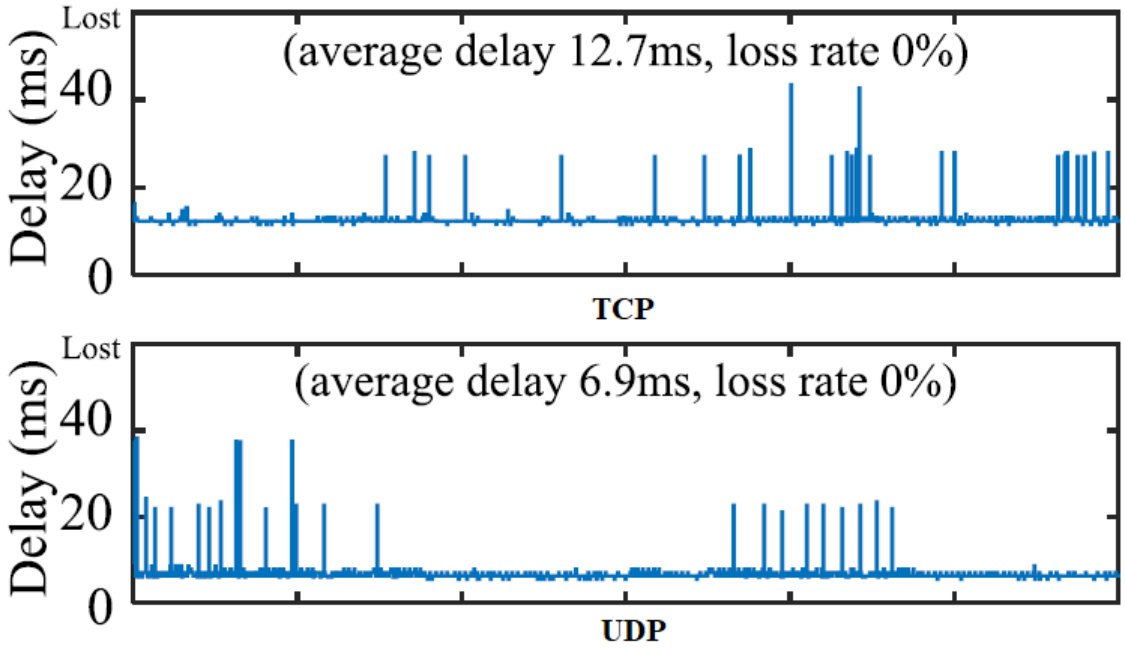

Figure 2. 5: Measured transmission delay for UDP and TCP [24]

On the other hand, Viktor Schreiber [25] has a different opinion from that of Yi Zhang [24]. Where he emphasises the advantages of UDP for distributed hardware in the loop simulation. He considers User Datagram Protocol (UDP) to be the most suitable option for distributed HiL due to high and networking over long distances capability.

Furthermore, [25] successfully implemented both open-loop and closed-loop testing setups, Figure 2.6 and Figure 2.7. For the open loop testing setup, the master host controls the slave host remotely. In this context, the slave host does not hold a virtual driver. Hence, the hosts are using the same inputs. However, all the hosts have their anti-lock braking system (ABS), hydraulic braking system (HBS), and vehicle controller (VC); respectively, the vehicle parametrization is identical.

For the closed-loop testing setup, the master host is controlling the slave host. However, the master and the slave are sharing the HBS and the ABS. More precisely, the master vehicle is using the responses of the slave HBS to perform the master ABS control. This exchange of states and data is executed in a closed loop. For this reason, the delay of the communication or RT system will influence the response of the system. [25]

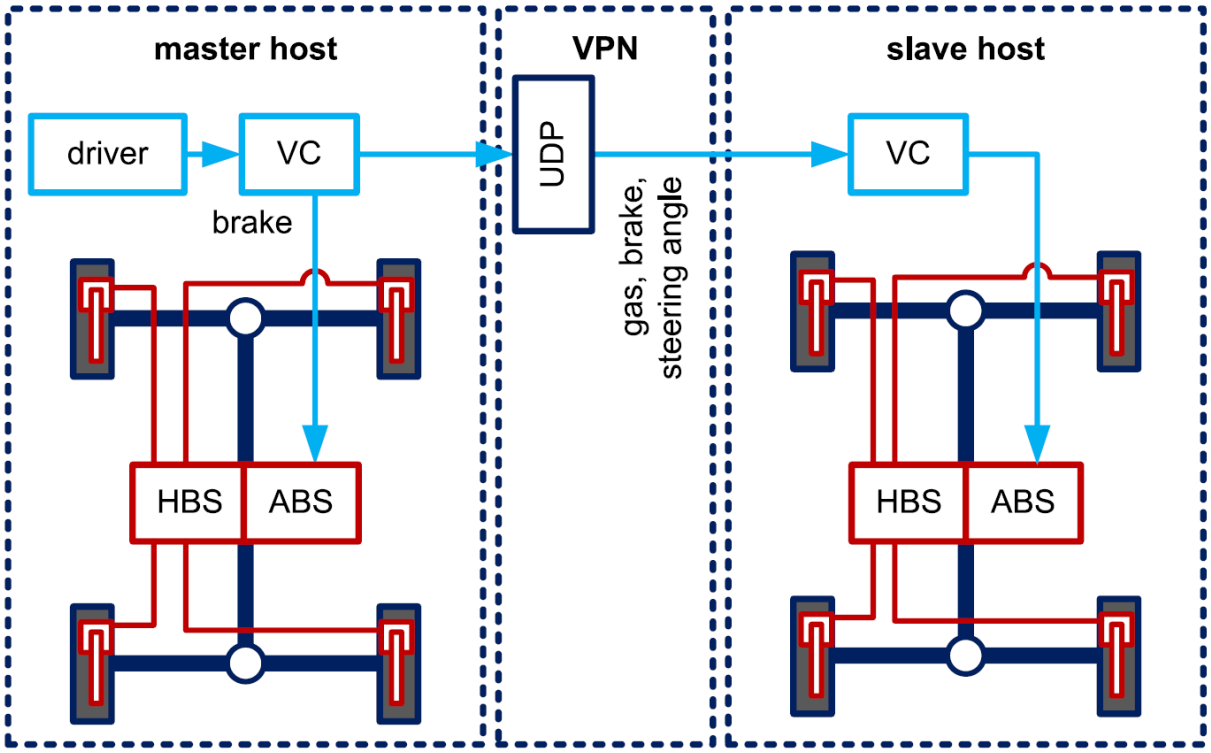

Figure 2. 6: Open-loop setup in [25]

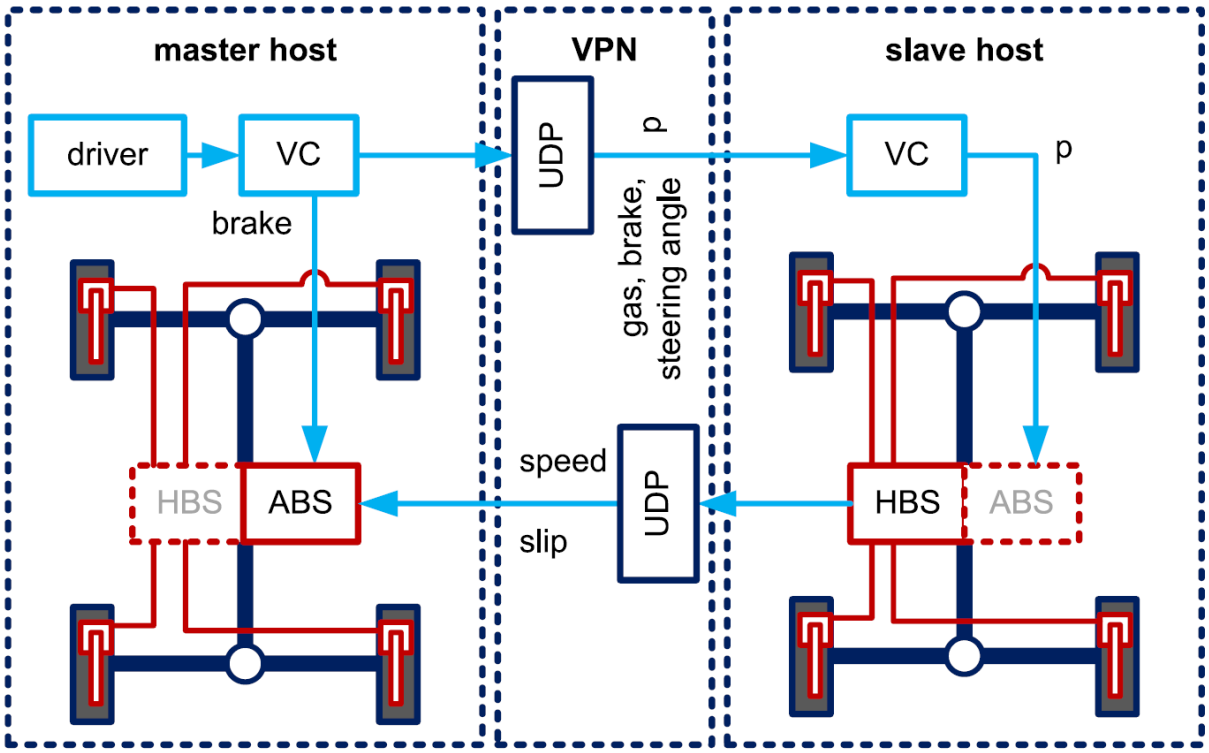

Figure 2. 7: Close loop setup in [25]

Despite satisfactory results in both phases of the project undertaken in [25], Schreiber acknowledges that the closed-loop distributed hardware in the loop simulation requires a more sophisticated communication architecture and simplified models to minimize the delay. The results presented in [25] emphasize the critical role of fast communication in facilitating distributed HiL simulations, particularly when real-time feedback is necessary. Delays, even if minor, can significantly impact the overall efficiency of these simulations. Thus, relying solely on UDP is not adequate, as other challenges may arise, such as packet loss, which necessitates a more sophisticated system to address these issues.

Moreover, in another research, Tulga [26] explains that “transaction-oriented mechanism that yields lower communication time delays albeit at the expense of a guarantee of delivery, ordering, or duplicate protection. This means that the packets may arrive out of order, in duplicates, or not at all, but as a benefit of this lack of network interface level error checks, the communication delay is reduced, which is more critical for the purposes of ID-HIL.”

Further in [26] the problem surrounding lost, out of order and duplicate packets is addressed by an event-based communication architecture that improves stability and transparency as well as synchronization related obstacles. The architecture is implemented in both nodes of the simulation, receiver, and transmitter end. Figure 2.8

represents the flow chart of the transmitter end and Figure 2.9 is the flow chart of the receiver end.

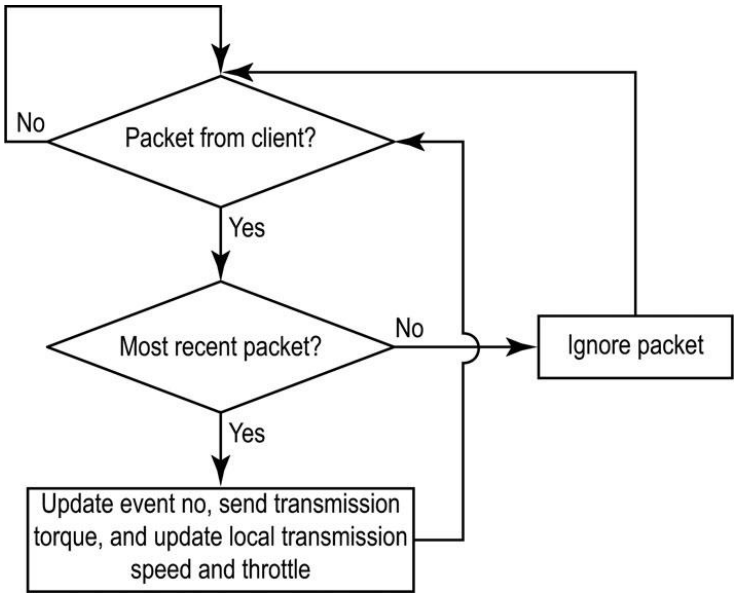

Figure 2. 8: Receiver end flow chart [26]

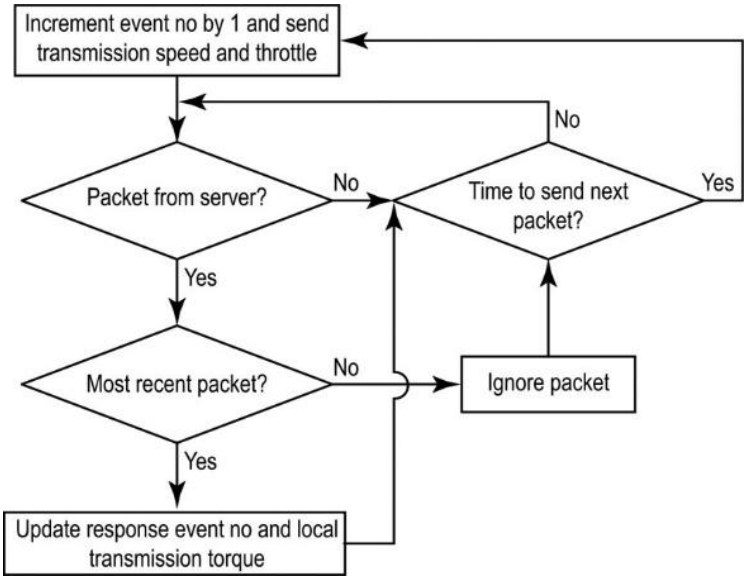

Figure 2. 9: Transmitter end flow chart [26]

A robust communication architecture, which employs the User Datagram Protocol (UDP), establishes a dependable and efficient communication infrastructure, thereby facilitating real-time, consistent distributed Hardware-in-the-Loop (HIL) simulations. Inspired by these studies, this thesis aims to conduct a feasibility study by developing a communication framework specifically tailored for electric drive systems.

2.2 Predictor

Additionally, in a more recent study [27], Schreiber used second-order Smith predictor to compensate for the delay in the communication between the systems. According to [28], “The traditional Smith predictor consists of a primary controller and a predictor structure. The idea behind the Smith predictor is that when the predicted output equals to the plant output, the signal from the plant model without delay can be used for control so that the controller can be designed without considering time delay and control performance can be significantly improved.”

Figure 2.10 represent the X-in-the-loop approach in [27]. In Schreiber novel approach, he uses three fundamental steps to assures the stability of the simulation. First of all, he underlined the significance of noise in coupled closed-loop systems. Schreiber believes digital filters such as IIR (infinite impulse response) and FIR filters (finite impulse response) can improve real-time capability, specific filter curves and linear phases.

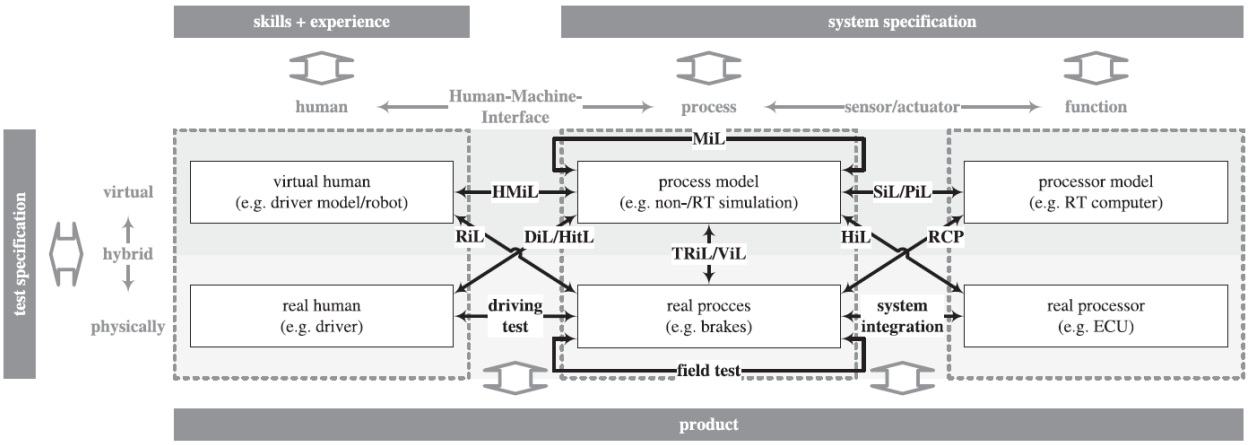

Figure 2. 10: X-in-the-loop setup [27]

Secondly, he uses smith predictor before feeding signals into the hardware to compensate for the communication delay. Finally, Schreiber process signals from the hardware through an observer, he claims that “An observation is a concept that estimates the actual system state using measured signals and a reduced order model. Disturbance like derived by noise and other influences are compensated by the observer and the model error is corrected”. [27] implements Kalman filter as an observer that mathematically estimates the variance from a linear time-invariant system. In [27] jitter has been neglected due to the high stability of the network.

The novel approach in [27] successfully undertook distributed HiL where the driver's response is the subjected signal that is a considerably slow dynamic. As [27] stated "A higher order of the predictor is required for highly dynamic systems. However, the choice of a higher order predictor is recommended with restrictions since each derivatisation induces numerical errors and noise. Smith predictors with lower order introduce instabilities which affects the native behaviour."

The discoveries of [31] in which smith predictor employed in the D-HIL simulation is emphasis the significance of using a predictor to improve the simulation robustness that is crucial in a D-HIL. Inspired by [31], this thesis utilizes a novel approach that is tailored around the characteristics of electric machine drive system, and unlike smith predictor that primarily works for linear systems, is capable of maintaining a stable real-time simulation environment for a dynamic system.

2.3 Summary:

Chapter two highlighted the significant role that HiL simulation is playing in automotive product development cycle and how introduction of ID- HiL can contribute to further improvement of time consumed to develop a new product, cost reduction, prototyping and early-stage integration of a system. The distributed hardware in the loop simulation trend witnessed in the automotive industry was ignited by Tulgo Ersal in [15] and Hosam in [23]. Perceptibly the communication aspect of distributing subsystems is the most notable difficulty in distributed hardware in the loop simulation, D-HiL.

Various factors can influence an ID- HiL including, the fidelity of the model and overall HiL setup. Also, coupling points, communication architecture, communication protocol and the use of observers and predictors are significantly influential.

In [24], [29] and [25], the impact of communication protocol was exploited. Despite the anthesis of opinions from [29] and [24], it can be concluded that TCP is the correct choice for slower systems. However, UDP is the most suitable protocol for D-HiL in more dynamic systems as it is more flexible, has a greater maximum transmission unit (MTU) and it benefits from low protocol overhead.

Moreover, communication architecture affects the results enormously. [26] has implemented an architecture that can compensate for the lost, duplicated or out of ordered packets when UDP is used. Also, it was discovered that the communication

challenges are far greater for a close-loop ID-HiL simulation and require more specific communication architecture and simpler model designing.

Even though the challenges with having an internet-distributed HiL setup are studied and acknowledged by the preceding research, and solutions based on the application have been proposed and, in some cases, validated, the nature of the electric drive system is introducing far more significant challenges. The obstacles are associated with the amount of data transfer, hence a greater bandwidth requirement as well as a tailored communication prerequisite to achieve swift and effective interaction between sub-systems. Furthermore, an electric drive system-specific observer and predictor are needed to have a reliable simulation setup.

This thesis is using a novel approach tailored around dynamic requirements of electric machine drive system, even though user datagram protocol is employed, the architecture of the communication, the prediction of errors and the approach to rectify them are all specific to this feasibility study. Another consideration is the dynamic and highly demanding nature of electric machine drive system that is completely different from the systems utilized in the preceding research.

This Chapter underlined the key areas of ID-HiL research in the automotive industry, highlighted the preceding research findings, and provided key elements that make this thesis distinguishable. It also exploited the foundation based on which this feasibility study's novel approach is constructed.
In the next chapter, insight into the EPS system’s electric machine and the reasons highlighting its importance to the automotive industry are reviewed. Moreover, the dSpace simulator is explained in more detail, along with a basic test that has been carried out and presented to display the functionality of the test setup.

Chapter 3

Technical Overview of the dSpace Scalexio Simulator and the Electric Machine Drive System Employed

This chapter explains the fundamentals of the system utilized in this thesis. It includes explaining the significance of electric machines in automotive and EPS systems, which are designated for this project. This chapter also provides further information regarding the real-time simulator (dSpace Scalexio).

Moreover, some basic tests have been undertaken to visualize the HiL simulation setup, and the results have been presented. This chapter is fundamental in understanding a technical overview of the configuration employed in this project and offers further insight into the test rig and subsystems that were previously discussed in Chapter One.

3.1 EPS system Electric Machine Overview (Unit Under Test)

Modernization in the structure of the automotive industry has resulted in the population of more electric machines in modern vehicles. There are more than 40 electric motors in a modern vehicle, some of which illustrated in Figure 3.1. They cover a wide range of applications, from lower power rating applications, such as seat motors, locks, and wiper motors, to higher power rating applications such as traction motors and electric power steering. [30]

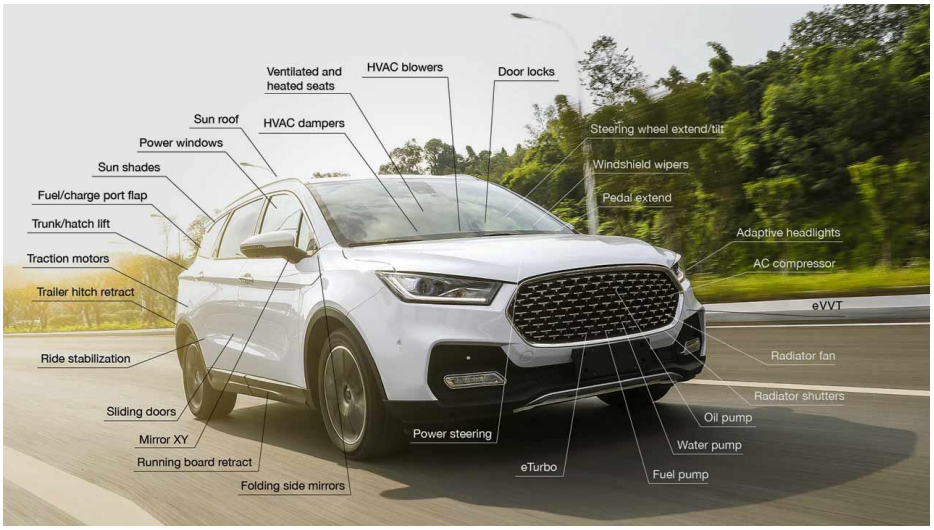

Figure 3. 1: Electric machine application in a modern vehicle [30]

The most common electric machine topologies in the automotive industry are induction machines, switched reluctance machines, and permanent magnet machines. Although permanent magnet machines are more expensive than the other topologies, manufacturers usually prefer them due to their high torque and power density and indisputable superior efficiency. [30]

The electric machine in this thesis is from an electric power steering (EPS) system. Electric machines were introduced to steering to supersede the conventional hydraulic power steering (HPS) system. Figure 3.2 depicts both EPS and HPS systems.

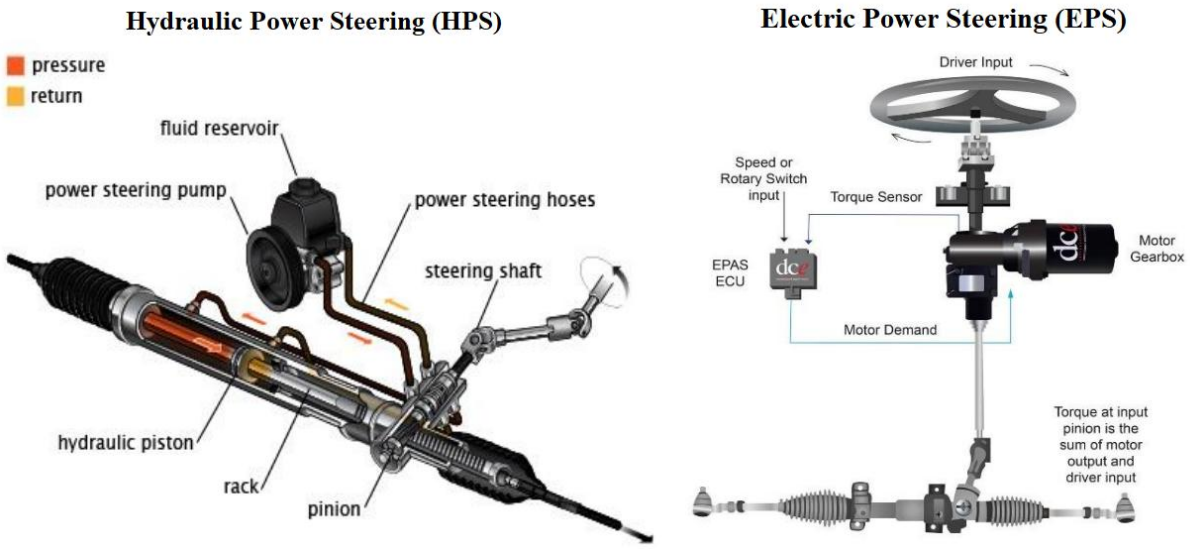

Figure 3. 2: Illustration of HPS and EPS systems [31]

In an EPS system, the power-assist is generated by an electric motor, and then a servo gear unit is used to deliver the generated force to the steering column or rack. Each EPS system consists of an electronic control unit, ECU, inside which power electronics are integrated. The power electronics can be used to actuate the electric motor. Once the desired torque is measured from the driver's steering command, ECU calculates the required power assistance and based on that, actuates the electric motor. [32]

The power demand dictates the type of electric motor in an EPS system and is different for every vehicle. Figure 3.3 represents power demand and characteristic curves of different electric motor topologies. Three common electric motor categories for EPS applications include DC motor, AC asynchronous motor, and AC synchronous motor. As shown in Figure 3.3, AC synchronous motors are the most common and effective electric motors for EPS applications. [31]

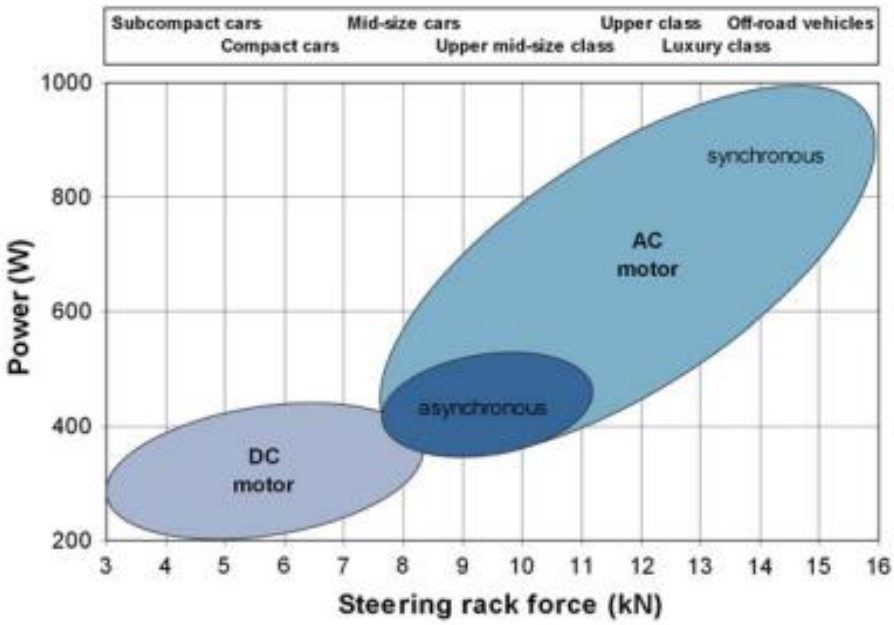

Figure 3. 3: Power demand and characteristic curves of different electric motors [31]

Synchronous machines contain no brushes, no slip rings, or mechanical commutators. In a synchronous machine, the speed of the machine is directly proportional to the frequency of the supply. The relationship between the speed and frequency of the power supply is shown in equation (1). [33]

$$f = \frac{N.p}{60} \qquad (1)$$

- N = Synchronous speed (Revolutions per minute)
- F = Frequency (Hertz)
- P = Pole pairs

Figure 3.4 represent the taxonomy of AC machines. Permanent magnet machines and reluctance machines are the two main types of synchronous machines.

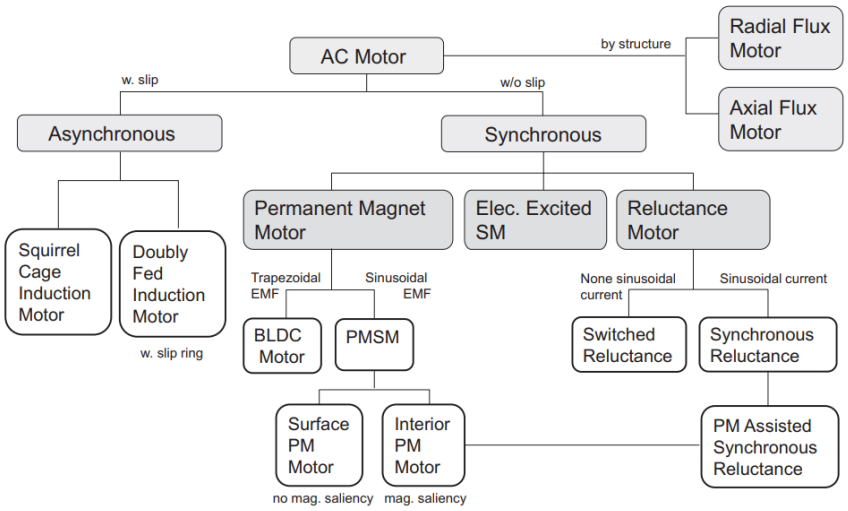

Figure 3. 4: AC machines classification [33]

A permanent magnet synchronous machine (PMSM) is the most commonly used machine topology, and the machine exploited in this project is also a PMSM. There are various types of permanent magnet machines. They can be divided into interior permanent magnet machines (IPM) and surface-mounted permanent magnet machines (SPM). The rotor is magnetized using permanent magnets instead of electromagnets in a permanent magnet machine. As the name implies, in an IPM, the permanent magnets are located inside the rotor, while in an SPM machine, the magnets are mounted on the surface of the rotor. Table 3.1 highlights some of the differences between the two kinds of PMSM. [33, 34]

	SPMSM	IPMSM
PM location	Surface	Cavities
PM fixation	Glue or band	Insertion
Field harmonics on PM	Large	Small
PM usage	Large	Relatively small
Saliency ratio	1	>1
Reluctance torque usage	No	Yes
Power density	Low	High
Speed range (field weakening)	Small	Large

Table 3. 1: SPMSM and IPMSM comparison [34]

SPM benefits from a more straightforward manufacturing process and shorter stack length, which allows shorter axial length. On the other hand, IPM has a higher torque density thanks to the rotor saliency. Torque equation for an IPM machine is presented

in equation (2), and similarly (3) represent torque equation for SPM machine. The presence of the two inductance terms in (2) accounts for reluctance torque that is produced due to varying winding inductance across the rotor position; this phenomenon is known as saliency or the rotor. The inductance term cancelling out in (3) highlights overall lower torque density of SPM machine. [33, 35]

$$T = \frac{3}{2}\frac{P}{2}(\psi_m + (L_d - L_q)i_{ds}^r)i_{qs}^r \qquad (2)$$

- T = Electromagnetic torque (Newton-meters, Nm)
- P = Number of poles
- ψ_m = Flux linkage (Weber-turns, Wb)
- I_q = Quadrature-axis current (amperes, A)
- I_d = Direct-axis current (amperes, A)
- L_d = Direct-axis inductance (Henries, H)
- L_q = Quadrature-axis inductance (Henries, H)

In SPM machine $L_d \approx L_q$, Therefore

$$T = \frac{3}{2}\frac{P}{2}\psi_m i_{qs}^r \qquad (3)$$

Moreover, SPM machine is not suitable for high-speed applications as there is a mechanical difficulty regarding the magnets; they cannot hold on to their position in high-speed operation, while their counterpart is ideal for high-speed applications as their magnets are secure inside the rotor structure. [33, 34]

As mentioned earlier, Newcastle University dedicated an 8-pole surface-mounted permanent magnet machine designed for an electric power steering (EPS) system to this project. The electric machine is further exploited in chapter four where a Simulink model of the drive system is developed.

This section provided a more detailed overview of the unit under test in this thesis. It also provided a broad overview of the EPS system and electric machine, which are fundamental to understanding the system employed for the feasibility study.

3.2 Real-Time HiL Simulator: dSpace Scalexio

As stated in Chapter Two, the real-time simulator is one of the main pillars of a real-time Hardware-in-the-Loop (HiL) simulation. Extensive use of HiL in various applications has resulted in an increasing number of real-time simulator manufacturers, including dSpace, National Instruments, AVL, Typhoon, etc.

dSPACE is a leading provider of hardware and software tools and solutions used to develop and test embedded control systems. The company was established in 1988, and its headquarters is located in Paderborn, Germany. The company has a global presence via its subsidiaries and offices in countries such as the United States, China, Japan, France, and the United Kingdom. The accessibility enables dSpace to serve a diverse and transnational customer base. [36]

Furthermore, dSPACE's simulators are not only widely used but also hold a crucial role in the global automotive industry. The dSpace simulators, a preferred choice of automotive manufacturers, suppliers, and research institutions worldwide, are instrumental in the design and testing of electronic control units (ECUs), autonomous driving systems, electric vehicle components, and other embedded systems. [36, 37]

Moreover, dSpace provides a diverse product range, such as hardware-in-the-loop (HIL) simulators, rapid control prototyping (RCP) systems, virtual validation solutions, automotive testing tools, and model-based development and simulation software. Newcastle University has a dSpace Scalexio model, consisting of three primary components: Scalexio LabBox, Scalexio rack system, and Scalexio processing unit. [36]

The Scalexio is a hardware-in-the-loop (HIL) simulation system commonly utilized in the automotive industry for testing and validation of embedded control systems. It is a modular real-time simulator that enables simulation of the behaviour of intricate systems, such as automotive electronic control units (ECUs), in a virtual environment. [37]

Furthermore, Scalexio contains hardware and software components. The hardware includes input/output (I/O) modules, processing units, and communication capabilities

of the system. These modules can be configured and interconnected to simulate the behavior of an actual system under test. The software component provides tools for model development, simulation, and real-time execution. [36]

The dSPACE Scalexio LabBox is a highly versatile and compact hardware-in-the-loop (HIL) system, Figure 3.5. Despite low I/O capability and dependency on an external processing unit, it offers exceptional flexibility and scalability in HIL simulation setups. In this thesis LabBox was only utilized for familiarization with dSpace. [36]

Figure 3. 5: Scalexio LabBox [37]

The Scalexio Rack System, as illustrated in Figure 3.6, is a more comprehensive version that is available in two models, each with a height of 9 or 12 units. This rack system includes various features, such as an integrated signal conditioning unit, an integrated failure insertion unit, and an integrated processing unit. Additionally, it allows for the inclusion of any dSpace I/O board, as listed in Appendix 12 and depicted in Figure 3.6. The rack system was used in different phases of this thesis, where external signals were fed into the model using the appropriate I/O board, and the processing unit was used to simulate different subsystems within the project. [36]

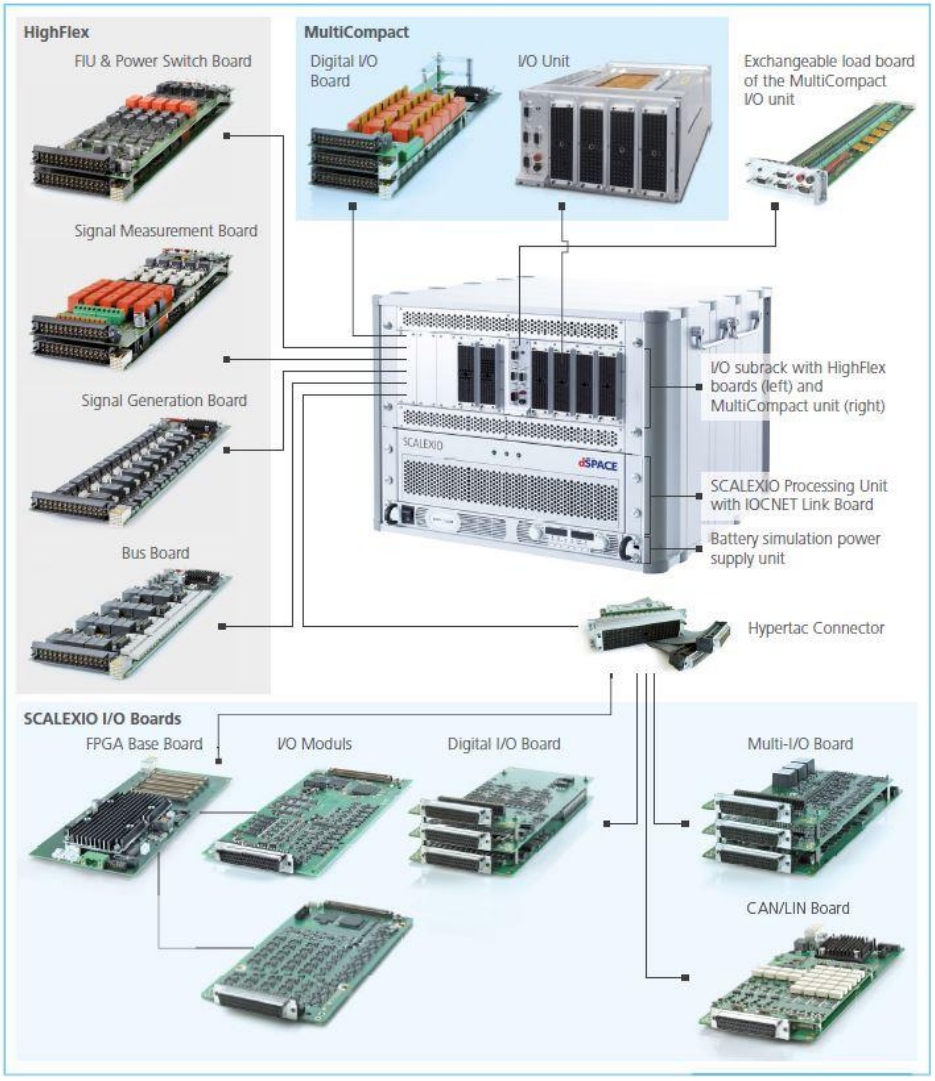

Figure 3. 6: Scalexio rack system and compatible I/O boards [36]

The Scalexio processing unit is equipped with a 4-core Intel processor, 16GB of RAM, and a 480GB solid-state disk (SSD). It supports both Ethernet and IOCNET, which is a communication medium specific to dSpace. This unit can be grouped with other processing units to form a multiprocessor system, enabling more complex models and enhancing computational capabilities. Within each Scalexio processing unit, one of the four cores is dedicated to services such as communication, while the remaining three cores are responsible for model computation. [36]

The Newcastle University dSpace Scalexio model, shown in Figure 3.7, benefits from having six processing units that form a multiprocessor system with a total of 18 cores available for model computation. This high number of cores allows for the handling of high-fidelity models. Furthermore, having multiple processing units enables the incorporation of a greater number of subsystems. In this project, the Scalexio rack

system was used for the interaction between hardware and software models, and the processing units were utilized to add different subsystems as needed.

Figure 3. 7: dSpace processing unit rack

This section gave an insight about the real-time simulator devoted to this project, the following section introduce an overview of the operation of the simulator and how processing of signal take place within a real-time HiL simulation using dSpace Scalexio.

3.3 dSpace Scalexio Principles of Operation

This section exploits the basic functionality of the dSpace Scalexio. Learning and utilizing the dSpace equipment was undoubtedly a massive burden on this research project and was highly time-consuming. As previously mentioned, two pieces of dSpace software, ConfigurationDesk and ControlDesk, were used. The two-software allowed for the configuration of communication between the Simulink model and the test rig and provided a user interface to monitor and record relevant signals.

Overall, ConfigurationDesk is the platform where the interfaces between setups, mode of operation, sampling time, and connection from hardware to the dSpace are

initialized. In addition, the ConfigurationDesk allows the user to specify the target remote location by setting a static IP address. Also, the software has the option to indicate the communication protocol. ConfigurationDesk also enables to specify the nature of communication between different models and hardware, as well as assigning different systems to specific processing units. Another option is the possibility of failure insertion, which was not exploited in this project. [36]

On the other hand, ControlDesk is a software that allows the user to manage and analyse the setup initiated in ConfigurationDesk. Within the software, specific parameters can be tuned and recorded over different periods of time. In addition, trigger conditions can be specified for recording certain parameters for a testing event. Test cases can also be configured, and communication statics can be monitored. Wireshark was used to analyse communication instead of the standard tools within the ControlDesk to achieve more precision. [36]

The project's first stage was doing preliminary tests, which included transmitting a voltage signal from a signal generator to the SPM Simulink model via dSpace Scalexio. In order to transform the sinusoidal waveform into a three-phase voltage prior to being sent to the SPM, a time delay block was included in the Simulink model. The time delay value was calculated using equations (4) and (5). The Simulink model shown in Figure 3.8 illustrates the use of dSpace input and output blocks, as well as time delay blocks, to generate sine waves that are 120 degrees apart. [38]

$$x(t) = A\sin(\omega t + \emptyset) \qquad (4)$$

- x(t): The instantaneous value of the waveform at time t.
- ω = The angular frequency in radians per second (ω = 2πf).
- A = The amplitude of the sine wave, indicating its peak value.
- Ø = The phase shift in radians.

$$t = \frac{\emptyset}{\omega} \qquad (5)$$

- t is the time (in seconds).
- f is the frequency (in hertz, Hz).

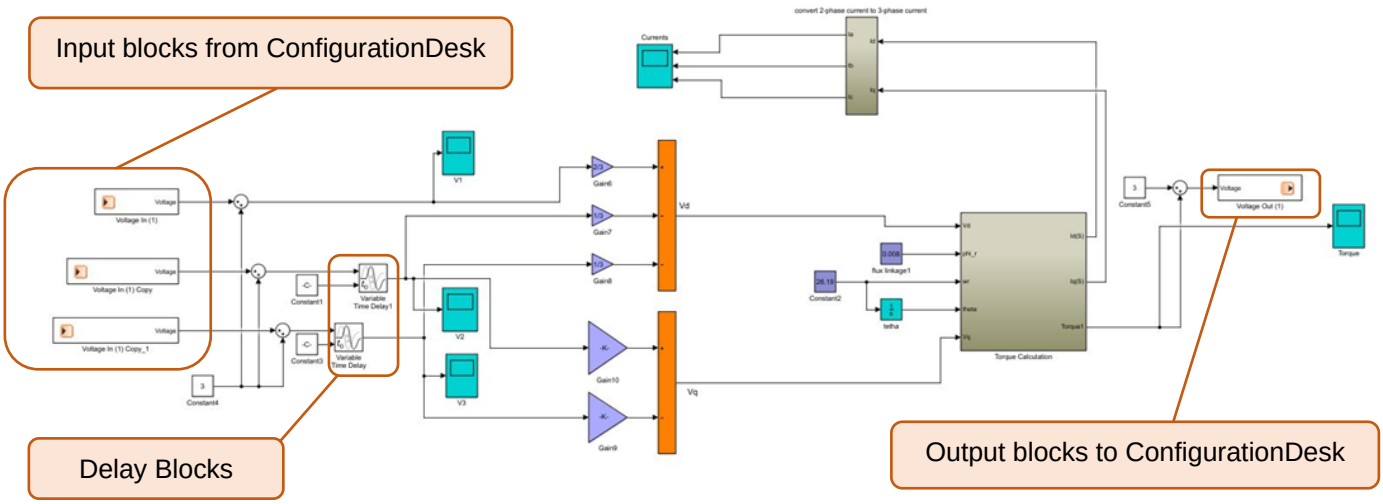

Figure 3. 8: Initial I/O testing with dSpace

The dSpace ConfigurationDesk is used to specify the external devices and other simulation aspects, such as communication type and sampling rate. It processes the Simulink model in C language and generates a .cdx file. This file is then imported into ControlDesk, a user interface provided by dSpace. ControlDesk allows users to monitor and record signals from various nodes of the Simulink model. Figure 3.9 illustrate the ConfigurationDesk window for this section.

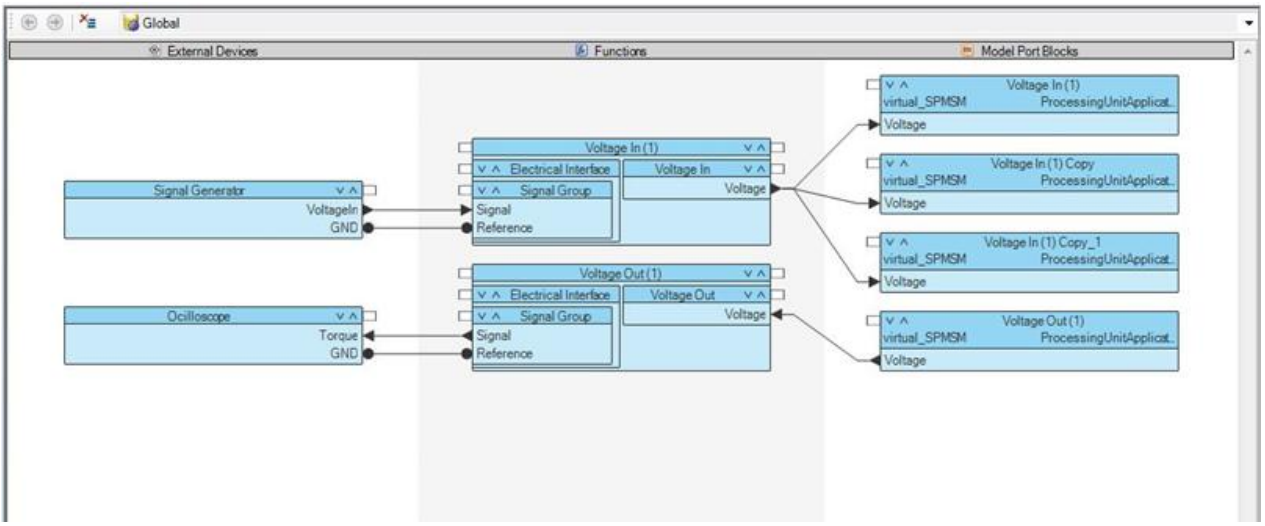

Figure 3. 9: Initial I/O test ConfigurationDesk environment

Subsequently, the torque curve that was obtained was shown on both ControlDesk, namely Figure 3.10, and also outputted from the dSpace Scalexio for visualisation on an oscilloscope (see Appendix 6).

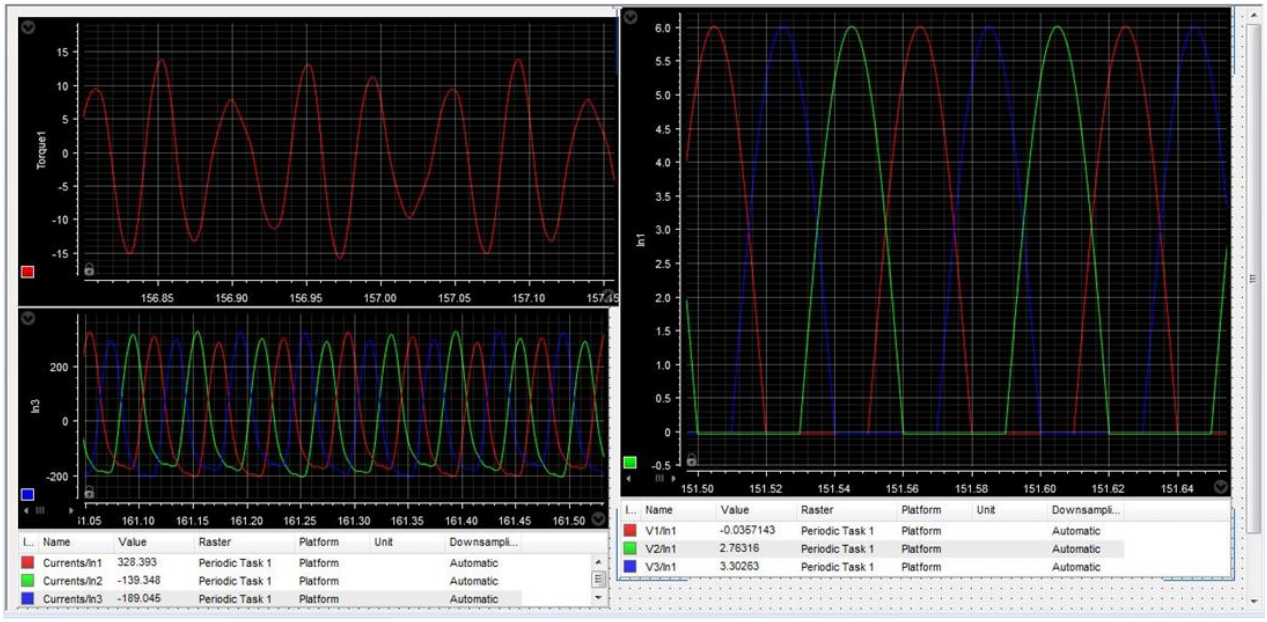

Figure 3. 10: ControlDesk visualization for basic I/O test

In this section, the real-time simulator employed in the study was further exploited. dSpace Scalexio is a highly capable real-time simulator and offers numerous I/O options that are fundamental for this study. Furthermore, this section elucidates the fundamental functions of the simulator, showcases its outcomes, and introduces the simulator's software, demonstrating its application in model integration.

3.4 Summary

This chapter provides a thorough introduction to electric machines in general, with a particular focus on the used SPM machine. Furthermore, the real-time simulator has been presented, accompanied with a comprehensive overview. The software linked to the real-time simulator was shown, and the interpolation between hardware and software was elucidated.

Furthermore, the operational principles of the simulator were elucidated, highlighting the utilisation of the device's I/O capabilities. Information on the available I/O boards on the simulator has been supplied, with more details accessible in Appendix 12.

The following chapter concentrates on the digital representation of the test rig, clarifying the dq model of the machine and modelling the controller and inverter. The developed model has been validated, and the findings have been provided.

Chapter 4

Modelling and Evaluation of Electric Machine Drive Systems

The previous chapter provided a technical overview of the real-time simulator, and the electric machine drive system employed. This chapter examines the development of the dq model for the drive system, which comprises three subsystems: the electric machine, the inverter, and the controller. The generated model was then validated, and the simulation results were compared with the real machine test results.

4.1 Electric Machine Drive System Simulink Model (DQ Model)

Creating a Simulink model of the drive system is a fundamental stage of the research project. Research into the stiffness and fidelity of the model along with type of solver is important, as they play a major role in the overall speed of the simulation. Therefore, the model has been developed taking these factors into account to achieve near real-world performance.

The drive system Simulink model developed in this section comprises three major components: an SPM machine, a controller, and an inverter. A set of parameters was retrieved from [13] and utilised in the modelling process, which are presented in Table 4.1. Additionally, further testing was conducted to derive flux linkage, which was essential to the model.

Parameter	Value
Number of Poles	8
DC Link Voltage	12 V
Base Speed	1000 rpm
PWM Carrier Signal	16 kHz
Maximum Phase Current	126 A
Phase Resistance	10.5 mΩ
L_s	65µH
Rated Torque	6.5 Nm

Table 4. 1: Electric machine drive system [13]

Additionally, the control mechanism employed in the test rig utilises field-oriented control (FOC), which will be elaborated upon in greater detail later in this chapter. It is also important to note that the models are designed to function in torque control mode. In this feasibility study, the motor was operated at base speed without the implementation of field weakening.

A series of tests were conducted to evaluate the performance of three different solvers, focussing on their accuracy and speed. Initially, the ode1 solver employing the Euler first-order formula for model computation was utilised. Equation (6) illustrates the ODE1 solver, where h is the step size, and f (t_n, y_n) is the function defining the derivative of y at time t_n. [39]

$$y_{n+1} = y_n + h.f(t_n, y_n) \qquad (6)$$

Furthermore, the fourth-order Runge-Kutta method, known as ode4, was investigated. Equation (7) serves as a representation of the solver, with h denoting the step size. The intermediate slopes, indicated by k_1, k_2, k_3, and k_4, are utilised to approximate the solution of the differential equation for a single time step. The formulas for the slopes can be found in Appendix 17. [39]

$$y_{n+1} = y_n + \frac{h}{6}(k_1 + 2k_2 + 2k_3 + k_4) \qquad (7)$$

Moreover, the ODE8 solver in MATLAB employs an 8th-order Runge-Kutta method, generally grounded in the Dormand-Prince formulation. The next step, y_{n+1}, is calculated through a weighted sum of intermediate slopes, equation (8). Where where h is the step size, k_i are evaluations of intermediate slopes presented in Appendix 17. Additionally, b_i, c_i, and a_{ij} represent particular coefficients utilised in the Dormand-Prince method. [39]

$$y_{n+1} = y_n + h\sum_{i=1}^{s} b_i k_i \qquad (8)$$

A fixed step sampling time of 62.5µs has been used that is equal to the model's switching frequency of 16kHz and simulation has taken place for 10 seconds. Figure 4.1 represent the torque signal of an SPM machine model running by a sinusoidal three-phase input voltage. Runge-Kutta and Dormand-Prince are showing almost an identical torque curve with a maximum difference of 0.033Nm, however Euler solver's resulted curve has a much greater difference to both other curves. Maximum of 1.976Nm to Dormand-Prince curve and 1.980Nm to the Runge-Kutta curve.

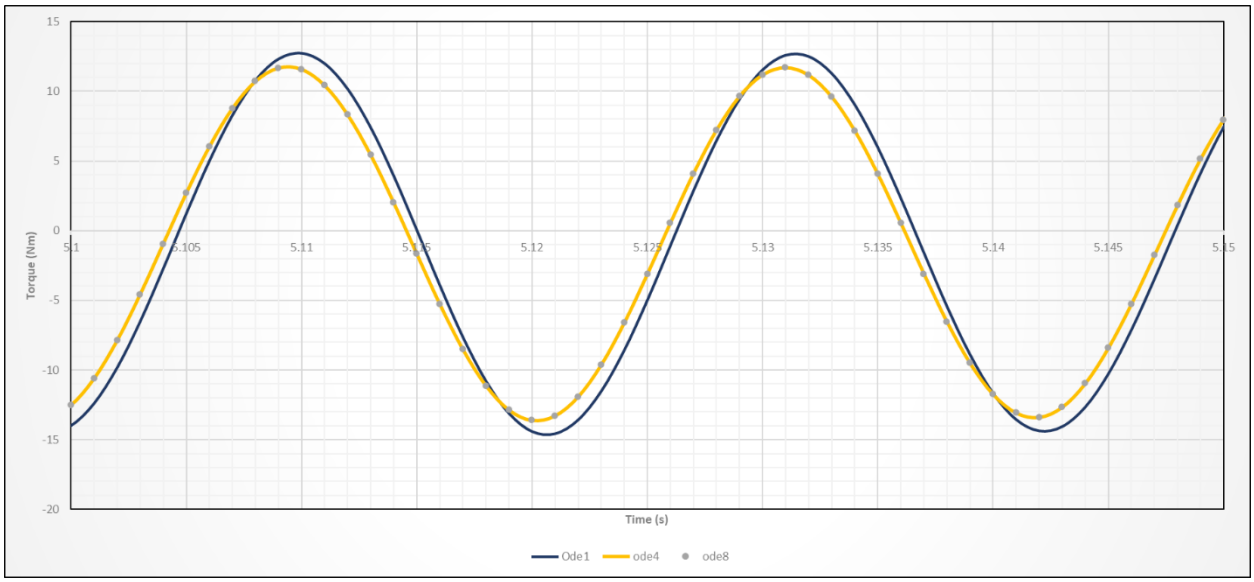

Figure 4. 1: Simulink solver comparison

The speed of the simulation is another crucial element, as emphasised in Table 4.2. The analysis indicates that the Dormand-Prince solver takes the longest to solve the same model, whereas the Euler method is the quickest of the three solvers. The Runge-Kutta solver demonstrates a speed advantage over the Ode8 solver by 60.3 seconds, while being only 13.8 seconds slower than the Ode1 solver. Furthermore, when ODE8 is utilised as a reference due to its status as the highest order solver, one can expect greater precision. The average error percentage is computed and displayed in Table 4.3, which shows that ODE45 closely aligns with ODE8 while still achieving a commendable speed. Conversely, ODE1 appears to be significantly misaligned, exhibiting an average error of 81.95 percent in comparison to ODE8. this indicates that it is considerably inefficient.

Solver	Time	Average Error %
Euler	9.1 seconds	81.95
Runge-Kutta	22.9 seconds	0.53
Dormand-Prince	83.2 seconds	-

Table 4. 2: Comparison of solver speed

Ultimately, ode4 (Runge-Kutta), a fourth-order solver, has been identified as the best option for this project because of its speed in calculations and precision during simulations.

The subsequent sub-sections will dive into the dq model employed in this project with greater detail, comprising three sections: the surface-mounted permanent magnet machine, the pulse width modulation and inverter, and the controller.

4.1.1 Surface Mounted Permanent Magnet Machine Modelling

In addition to the explanations in chapter three on permanent magnet synchronous machines (PMSM), in this section more detail on the approach to the dynamic modelling of the electric machine is provided.

The classification of PMSMs is primarily based on the positioning of the PM within the rotor, leading to two main categories. When the permanent magnets are installed on the surface of the rotor, the motor is referred to as surface mounted PMSM, or SPM for short. Alternatively, when permanent magnets are integrated into the rotor core, the motor is designated as an interior permanent magnet synchronous motor or IPM. Magnetic anisotropy arises from PM embedding, leading to a difference between the voltage and torque equations. The cross-section of the most prevalent IPM and SPM topologies is represented in Appendix 1. [33]

As indicated previously, the PMSM in this project is an SPM. To create the dynamic model of the SPM, dynamic equation of the machine needs to be derived. Flux linkage of a PMSM is produced by the stator current and the rotor permanent magnets. It must be noted that flux linkage is a function of the rotor position, θ, and it changes as the rotor spins. [33]

Figure 4.2 depicted how the permanent magnet flux linked to each phase of the magnetic flux from the stator winding changes as the rotor position changes. Figure 4.2 also indicates that flux linkage is maximum at 0^o and zero at 90^o. consequently, fundamental component can be described as a cosine function where PM flux, ψ_m, is a constant. [40]

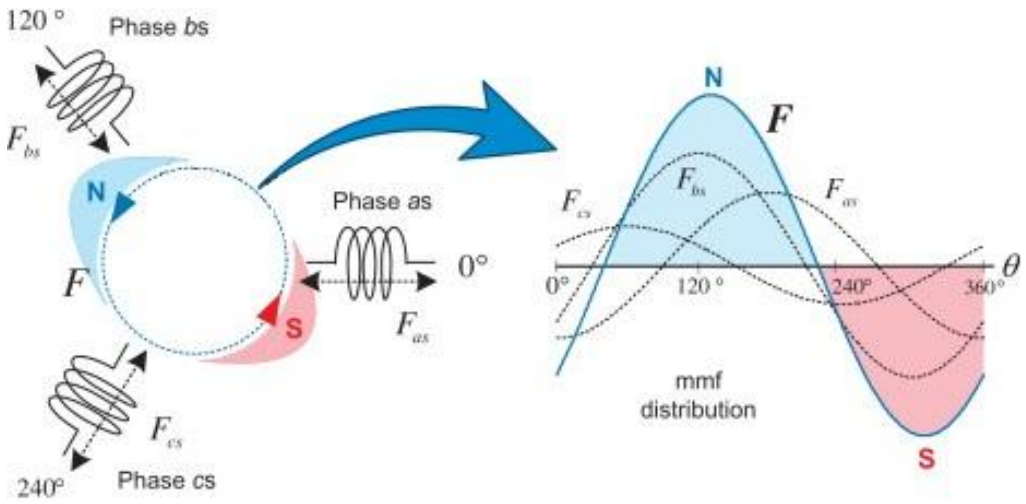

Figure 4. 2: Magnetomotive force (MMF) distribution [40]

Considering the 120º phase shift between the three-phase magnetic field produced by the stator windings. Flux linkage in rotary reference frame can be calculated for an SPM from equation (9).

$$\lambda_{dq}^{r} = L_s i_{dq}^{r} + \psi_m \qquad (9)$$

Considering the dynamic stator equivalent circuit of both q-axis, Figure 4.3, and d-axis, Figure 4.4, where core losses have been neglected.

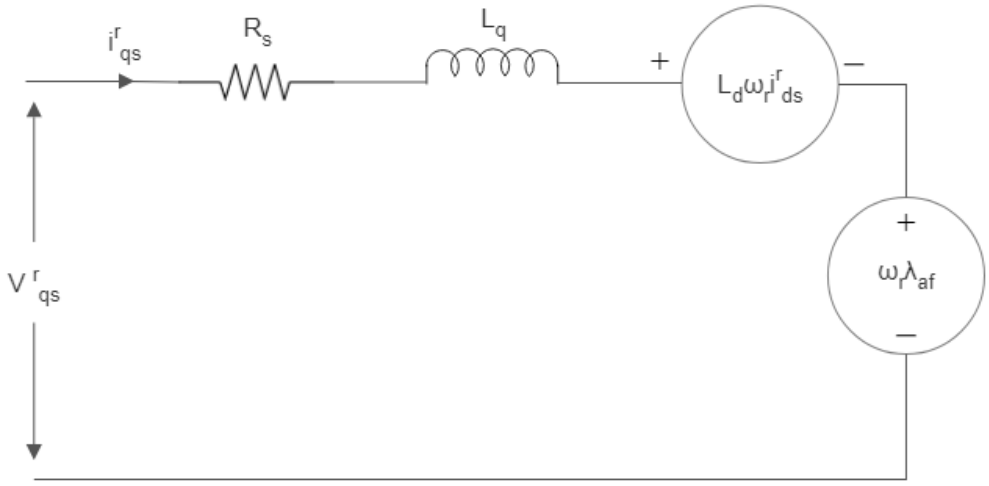

Figure 4. 3: Dynamic stator q-axis equivalent circuit of PMSM [41]

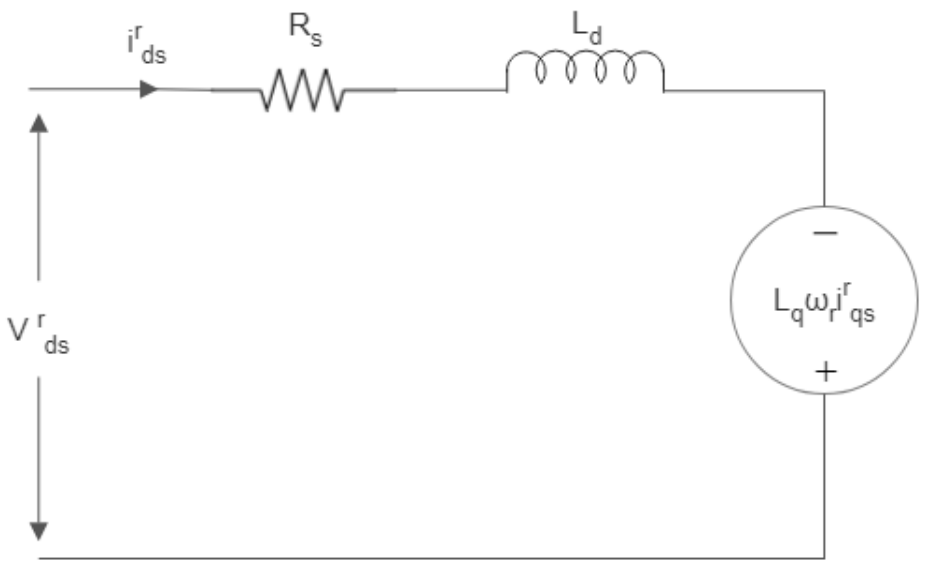

Figure 4. 4: Dynamic stator d-axis equivalent circuit of PMSM [41]

The SPMSM dynamics in rotary dq reference frame is provided by equation (10) and equation (11), where $\omega\psi_m$, refers to the induced back-emf and $\omega L_s i_{dq}^r$ is the coupling voltage resulted from transforming into rotating reference frame. [35, 41]

$$v_d^r = r_s i_d^r + L_s \frac{d}{dt} i_d^r - \omega L_s i_q^r \qquad (10)$$

$$v_q^r = r_s i_q^r + L_s \frac{d}{dt} i_q^r + \omega L_s i_d^r + \omega\psi_m \qquad (11)$$

Thus, the differential equation that is fundamental for the modelling of the SPMSM is given by equation (12) and equation (13).

$$\frac{d}{dt} i_d^r = \omega . i_d^r - \frac{r_s}{L_s} i_d^r + \frac{v_d^r}{L_s} \qquad (12)$$

$$\frac{d}{dt} i_q^r = -\omega . i_d^r - \frac{r_s}{L_s} i_q^r + \frac{v_q^r}{L_s} - \frac{\psi_m . \omega}{L_s} \qquad (13)$$

Furthermore, torque is described to be a cross product of the flux linkage and q-axis current and is calculated from equation (14).

$$T = \frac{3}{2}\frac{P}{2}\psi_m i_q^r \qquad (14)$$

Figure 4.5 illustrates the Simulink blocks used for torque calculation in the SPM, which will be utilised and validated later in this chapter within the complete drive system model.

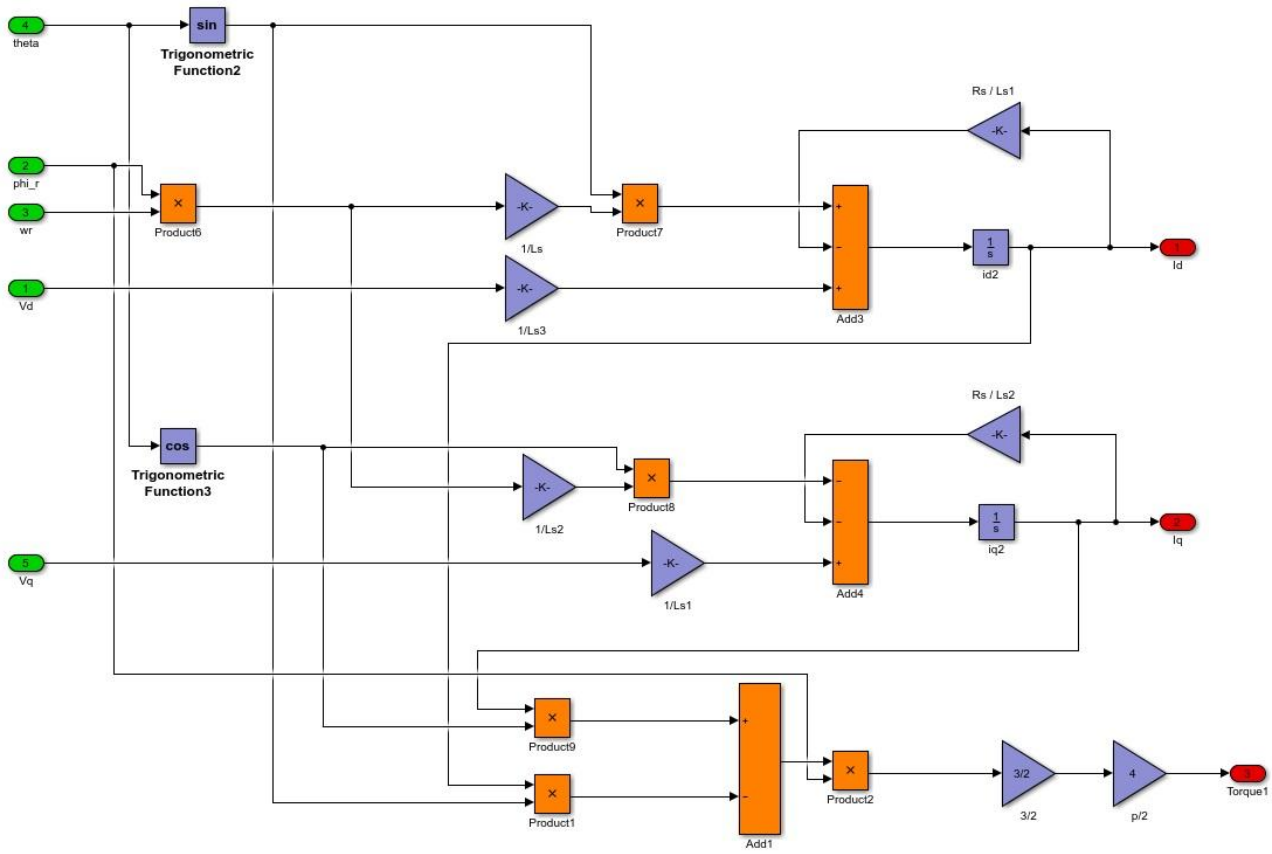

Figure 4. 5: SPM Simulink blocks

Despite retrieving most of the parameters from [13]. Nonetheless, the value of flux linkage remained unidentified. Consequently, a no-load test was conducted in which the SPM machine being evaluated was connected to a dynamometer operating at a steady speed of 1020 revolutions per minute to monitor and document the line-to-line back-emf. The phase connector pins at the back of the SPM are illustrated in Appendix 15.

It is important to note that fitting the electric motor to the test rig required some mechanical adjustments. This was made possible by the support of ZF Automotive Ltd, who generously provided the motor and testbed, as well as the technicians in the Merz Court mechanical lab, who assisted in creating a plate and setting up the test rig. Appendix.3 illustrates the plate drawing.

The control board utilised for this project is the Texas Instrument F28377D. The control script and the graphical user interface (GUI), as detailed in Appendix 16, were provided by Dr. Dave Atkinson. The test rig was operated using Visual Studio and Coder to load the software onto the control board, while a GUI designed in MATLAB facilitated the

operation of the electric motor test rig. Figure 4.6 illustrates the measured back-emf voltage at a speed of 1020 revolutions per minute.

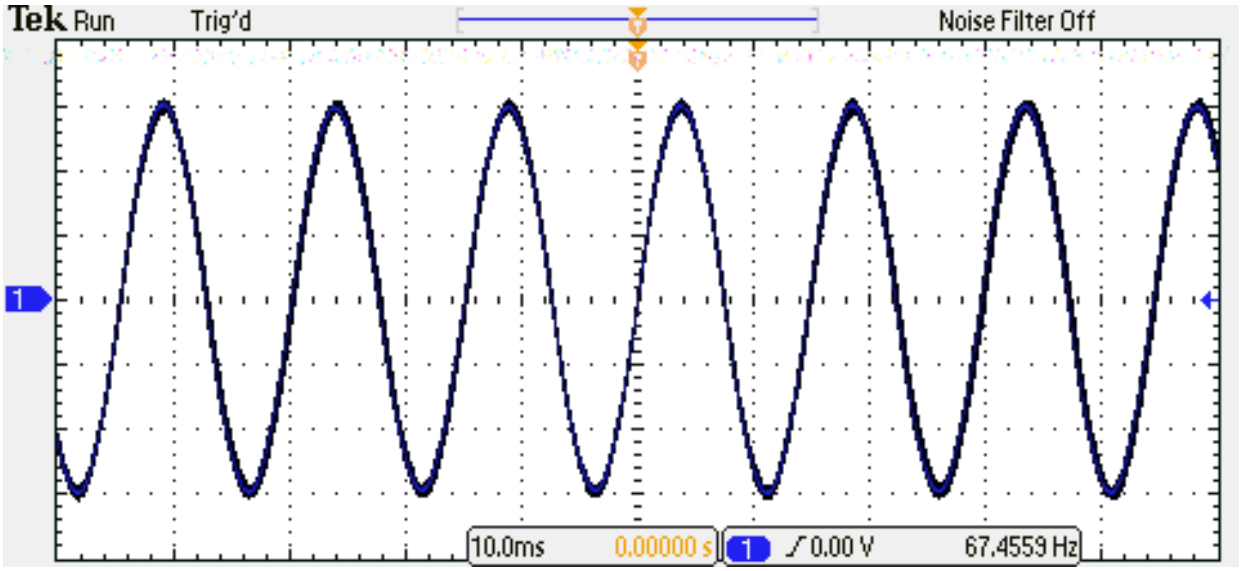

Figure 4. 6: Line-line back emf

At no-load resistance is considered to be zero and it has been assumed that the emf waveform is a perfect sinewave. Hence Emf = E_{max} sin(ω.t) and Emf = R.i + $\frac{d\psi}{dt}$, where R = 0. Therefore, the equation can be rearranged as $\psi = \int_0^t Emf$ which can be written as $\psi = Emf \int_0^t \sin(\omega.t)$. Taking the integration results in $\psi = \frac{-Emf}{\omega} \cos(\omega.t)$. Since the maximum Emf is used, the term ω.t is one and the cosine results in 0. Consequently equation (15) can be used to calculate the flux linkage to be 8.024 mV.s.

$$\psi = \frac{Emf}{\omega} \qquad (15)$$

Once all parameters are fully defined, the development of the Simulink model can proceed. The model operates below base speed, and the mechanical section of the machine has been analysed using the machine equation of motion, specifically equation (16), where J represents inertia, and B denotes the viscous friction coefficient.

$$J\frac{d\omega}{dt} + b\omega = T_s(t) \qquad (16)$$

This section provided an in-depth overview of the Simulink representation of the SPM machine, highlighting the derivation of specific parameters from measurements that were then employed in the development of the machine's dynamic Simulink model.

This section will detail the development of the Pulse Width Modulation and Inverter Simulink model, outlining its various components and functions.

4.1.2 Pulse Width Modulation and Inverter

This section begins with a comprehensive overview of the inverter and elucidates the application of the PWM technique in the operation of the switching device. Figure 4.7 illustrates the Simulink model of the inverter, which comprises a three-phase voltage source inverter featuring six switches and a shared DC link. The inverter comprises three switching arms: the first arm consists of T1 and T2, the second arm includes T3 and T4, and the third arm is made up of T5 and T6. Various physical switching devices exist, tailored to specific applications. Power MOSFETs, or metal–oxide–semiconductor field-effect transistors, are suitable for applications that require low voltage and high frequency. Moreover, insulated-gate bipolar transistors (IGBTs) are frequently utilised in medium and high-power inverters, while field-effect transistors (FETs) are applied in low voltage or high frequency scenarios. [35]

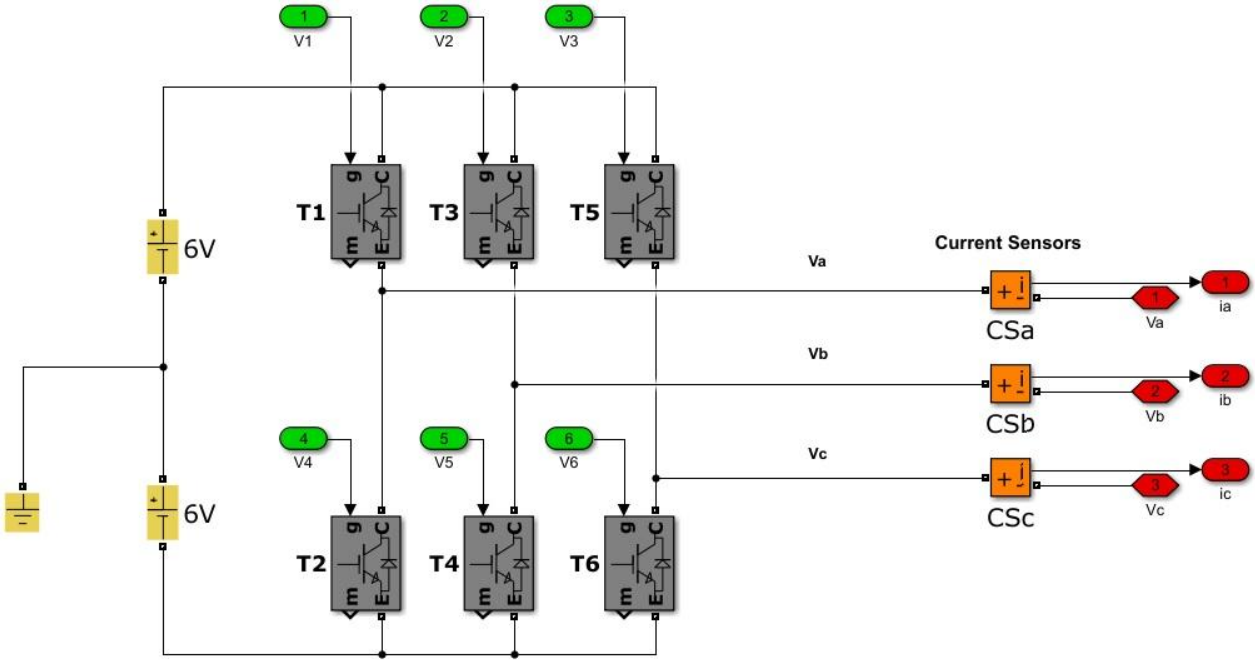

Figure 4. 7: Voltage source inverter with six switching IGBT and three current sensors

The DC voltage in a voltage supply inverter can be from an AC source which requires a rectifier or from a DC source as presented in Figure 4.7. in addition, each power switch needs a gate drive circuit that acts as power amplifier, it improves the low-power input from the controller and generates the appropriate high current gate drive for the

power switching device. The switching pattern of the power devices are dictated by Pulse Width Modulation (PWM).

There are several PWM techniques, including selective harmonic elimination PWM, sinusoidal PWM, and space vector modulation (SVM). The output voltage of SVM is 15% greater than that of sinusoidal PWM. In this project SVM is used as the switching technique for the inverter. Figure 4.8 illustrates that SVM functions as a form of pulse width modulation; however, in contrast to phase leg modulation, it relies on a vector representation of the three-phase voltages. [42]

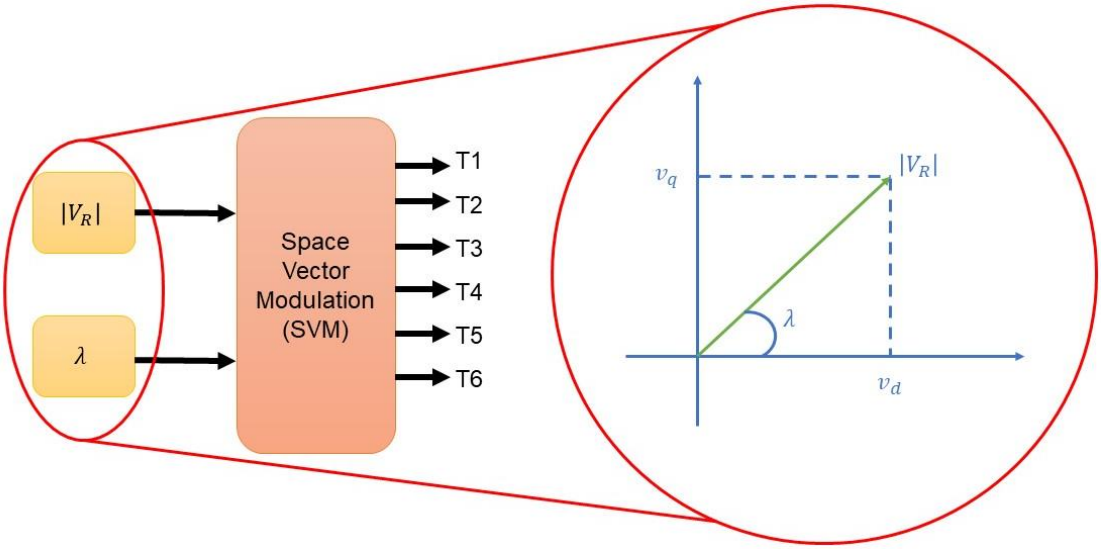

Figure 4. 8: Space vector modulation and input demand voltage vector

The Simulink model of the controller, illustrated in Figure 4.9, takes in a three-phase current from the machine along with the rotor angle theta. Utilising these parameters in conjunction with Clarke and Park transformation allows for the calculation of currents in the d and q axis rotary frame. The modulator takes in the required stator voltage, presented in either rectangular or polar format, and generates switching signals T1 to T6 for the six inverter switching devices. Equation (17) provides both rectangular and polar form of the reference stator voltage vector V_R.

$$\overline{V_R} = |V_R| \exp(j\lambda) = v_d + jv_q \qquad (17)$$

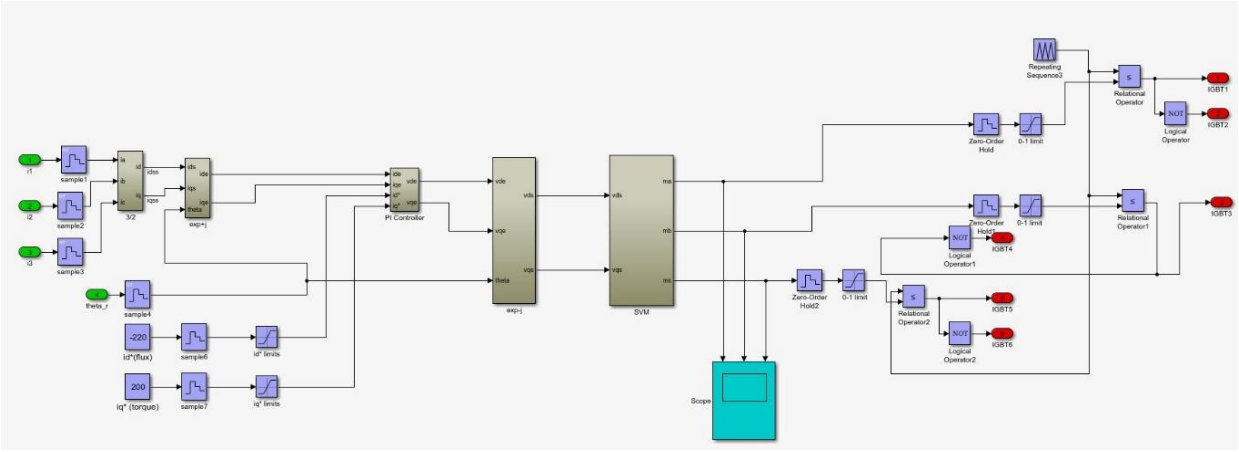

Figure 4. 9: Controller Simulink blocks

The inverter is associated with eight distinct switching states. Table 4.4 illustrates the line and phase voltages associated with each switching state of the three-phase inverter, while Figure 4.10 shows the corresponding switching states. The static magnetic field generated by each switching state (assuming that the states produce a DC current flow) dictates the direction of the vector.

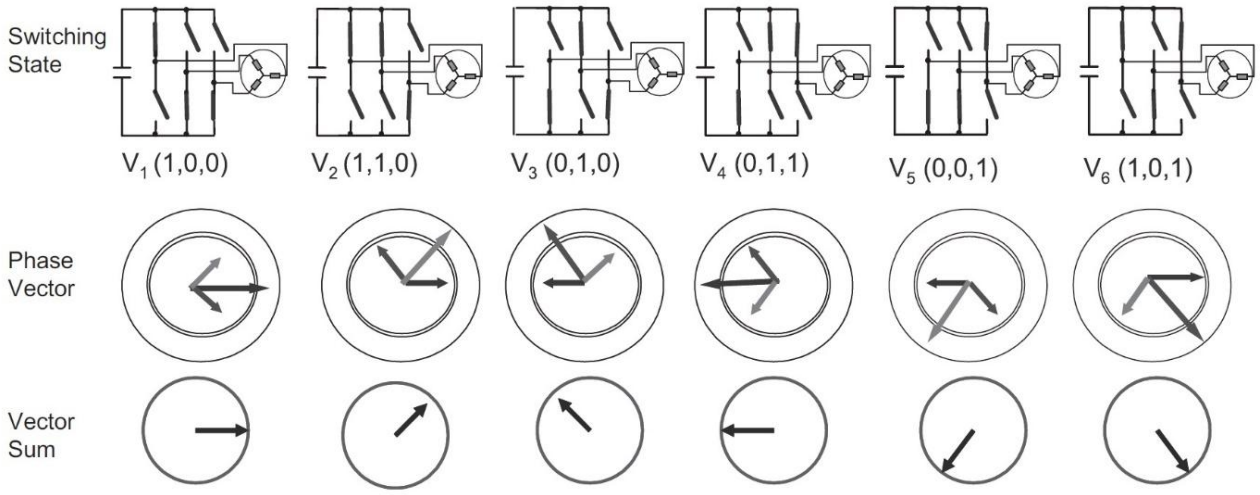

Figure 4. 10: Switching states, phase, and sum vectors [35]

State	Vector	Line Voltage			Phase Voltage			2-axis Voltage	
		$\frac{V_{ab}}{V_{dc}}$	$\frac{V_{bc}}{V_{dc}}$	$\frac{V_{ca}}{V_{dc}}$	$\frac{V_{an}}{V_{dc}}$	$\frac{V_{bn}}{V_{dc}}$	$\frac{V_{cn}}{V_{dc}}$	$\frac{V_d}{V_{dc}}$	$\frac{V_q}{V_{dc}}$
000	$\bar{V}_0$	0	0	0	0	0	0	0	0
100	$\bar{V}_1$	1	0	-1	$\frac{2}{3}$	$-\frac{1}{3}$	$-\frac{1}{3}$	$\frac{2}{3}$	0
110	$\bar{V}_2$	0	1	-1	$\frac{1}{3}$	$\frac{1}{3}$	$-\frac{2}{3}$	$\frac{1}{3}$	$\frac{1}{\sqrt{3}}$
010	$\bar{V}_3$	-1	1	0	$-\frac{1}{3}$	$\frac{2}{3}$	$-\frac{1}{3}$	$-\frac{1}{3}$	$\frac{1}{\sqrt{3}}$
011	$\bar{V}_4$	-1	0	1	$-\frac{2}{3}$	$\frac{1}{3}$	$\frac{1}{3}$	$-\frac{2}{3}$	0
001	$\bar{V}_5$	0	-1	1	$-\frac{1}{3}$	$-\frac{1}{3}$	$\frac{2}{3}$	$-\frac{1}{3}$	$-\frac{1}{\sqrt{3}}$
101	$\bar{V}_6$	1	-1	0	$\frac{1}{3}$	$-\frac{2}{3}$	$\frac{1}{3}$	$\frac{1}{3}$	$-\frac{1}{\sqrt{3}}$
111	$\bar{V}_7$	0	0	0	0	0	0	0	0

Table 4. 3: Switching states and associated line voltages, phase voltages, and 2-axis voltages.

Since there are six switches in total throughout each period, this switching pattern is referred to as six-step operation. Each of the six inverter switching states corresponds to one of the six sections of a hexagon, depicted in Figure 4.11. As noted in Table 4.4, V_0 (all upper-level switches are closed) and V_7 (all lower-level switches are closed) is when the motor terminal voltage is zero, they are known as space vectors and their direction is like generated flux in DC state.

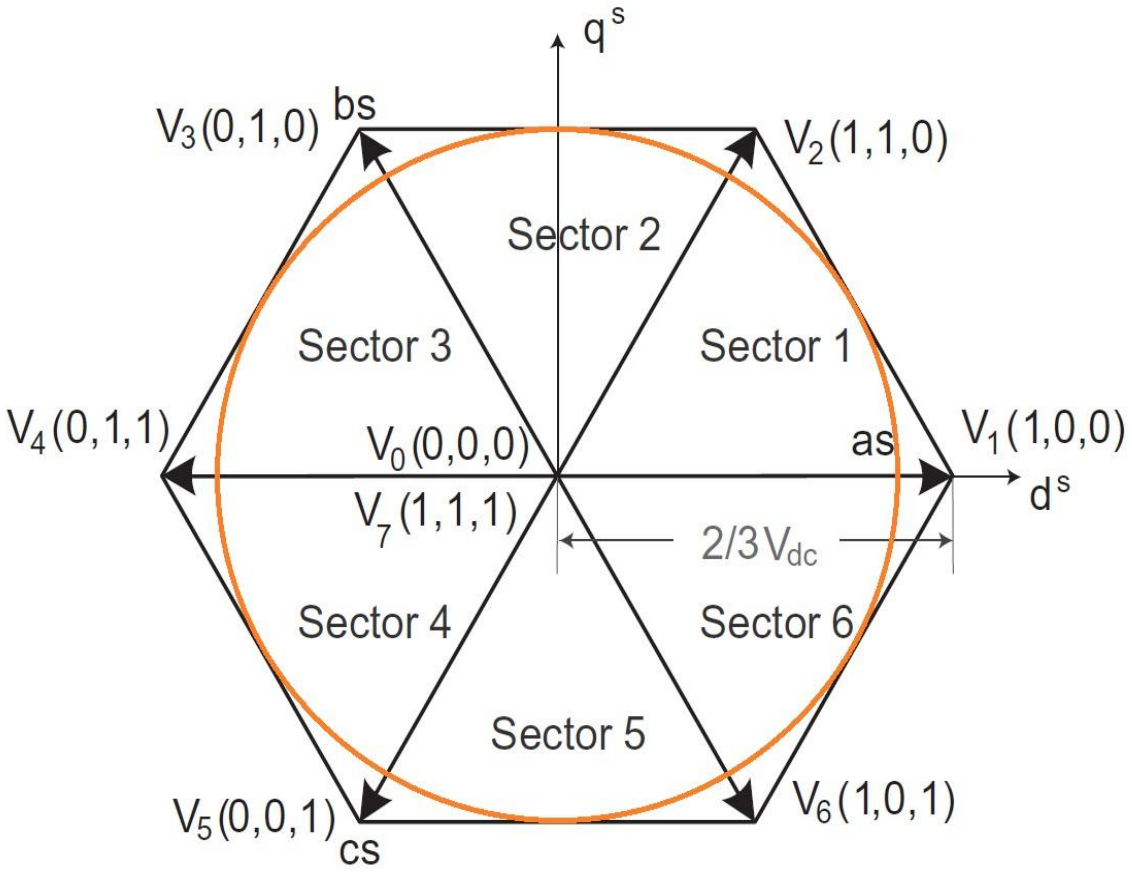

Figure 4. 11: Space vector diagram [35]

Figure 4.11 illustrates the division of the hexagon into six sections, each consisting of six state vectors, with a 60-degree separation between each pair of vectors. The VR vector is capable of positioning itself at any location within the hexagon during standard operation. During steady state, the voltage vector exhibits a consistent rotation around the hexagon, maintaining a constant magnitude and angular velocity. The direction of rotation determines the orientation in which the machine is spinning. The orange circle within the hexagon signifies the trajectory of the maximum voltage vector when sinusoidal modulation is applied. This thesis does not delve into the techniques of overmodulation, which facilitate the expansion of the circle and enable higher output voltages. [43]

By analysing the hexagon and considering the reference voltage between state vector V1 and V2, as shown in Figure 4.12, the duration for which the two state vectors are applied in a PWM cycle to reach the reference voltage can be calculated by understanding the magnitudes of components Ua and Ub.

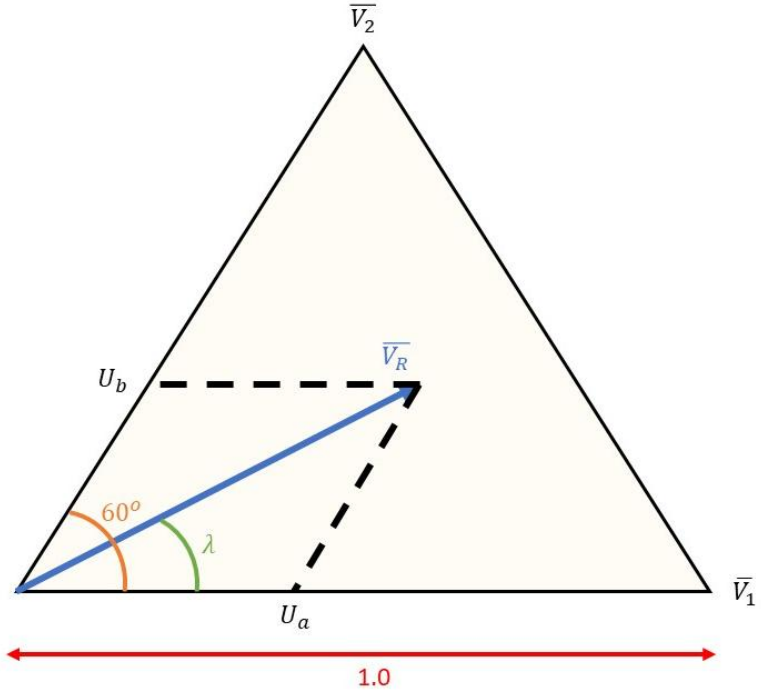

Figure 4. 12: Projections of the reference voltage vector onto two adjacent state vectors

The mapping of V_1, V_2 and V_R, allows for calculation of U_a and U_b with equation (18) and equation (19). (Refer to Appendix 4 for derivation)

$$U_a = \frac{2}{\sqrt{3}}|\overline{V_R}|\sin(60^o - \lambda) \qquad (18)$$

$$U_b = \frac{2}{\sqrt{3}}|\overline{V_R}|\sin(\lambda) \qquad (19)$$

Taking the calculated components into account, the timing of the leg switching is as noted in equation (20) to equation (22). A single PWM cycle illustrated in Figure 4.13, shows the timing of the PWM when Reference vector is in the first sector between V_1 and V_2.

$$T_a = T.U_a \qquad (20)$$

$$T_b = T.U_b \qquad (21)$$

$$T_c = T - T_a - T_b \qquad (22)$$

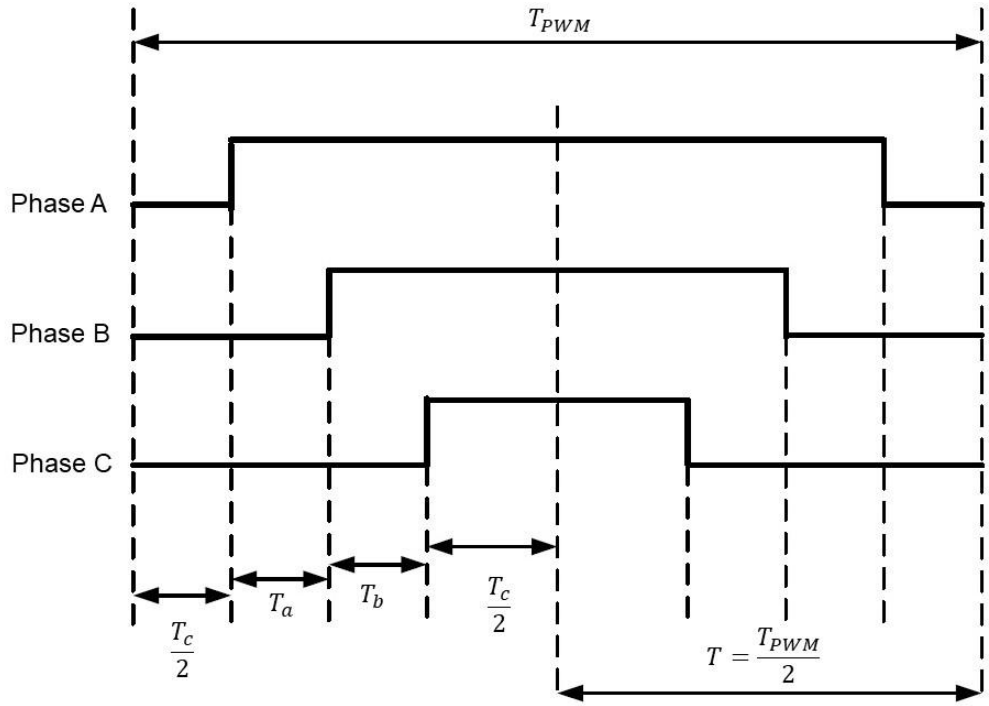

Figure 4. 13: Timing of inverter leg switching signals.

By utilising the PWM, one can determine the modulation indexes per unit for each phase through the application of equation (23) for phase A, equation (24) for phase B, and equation (25) for phase C. The digital counter integrated within a microcontroller facilitates the execution of PWM timing across the inverter phases. Figure 4.14 illustrates the comprehensive overview of the switching pattern development for the inverter. Moreover, figure 4.15 demonstrates the measured modulation signals obtained from the Simulink model. Simulink blocks are detailed in Appendix 5.

$$miA = \frac{2T_A + 2T_B + T_C}{T_{PWM}} \quad (23)$$

$$miB = \frac{2T_B + T_C}{T_{PWM}} \quad (24)$$

$$miC = \frac{T_C}{T_{PWM}} \quad (25)$$

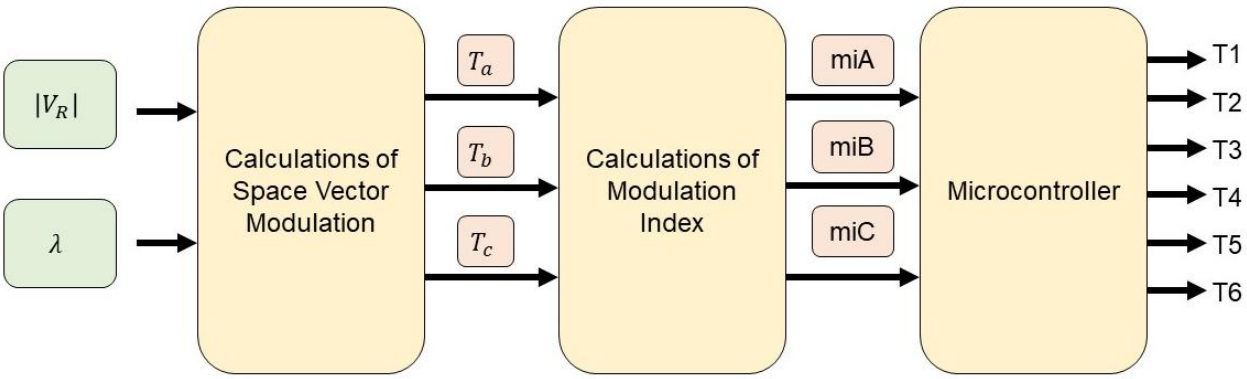

Figure 4. 14: Inverter switching pattern calculation and implementation steps.

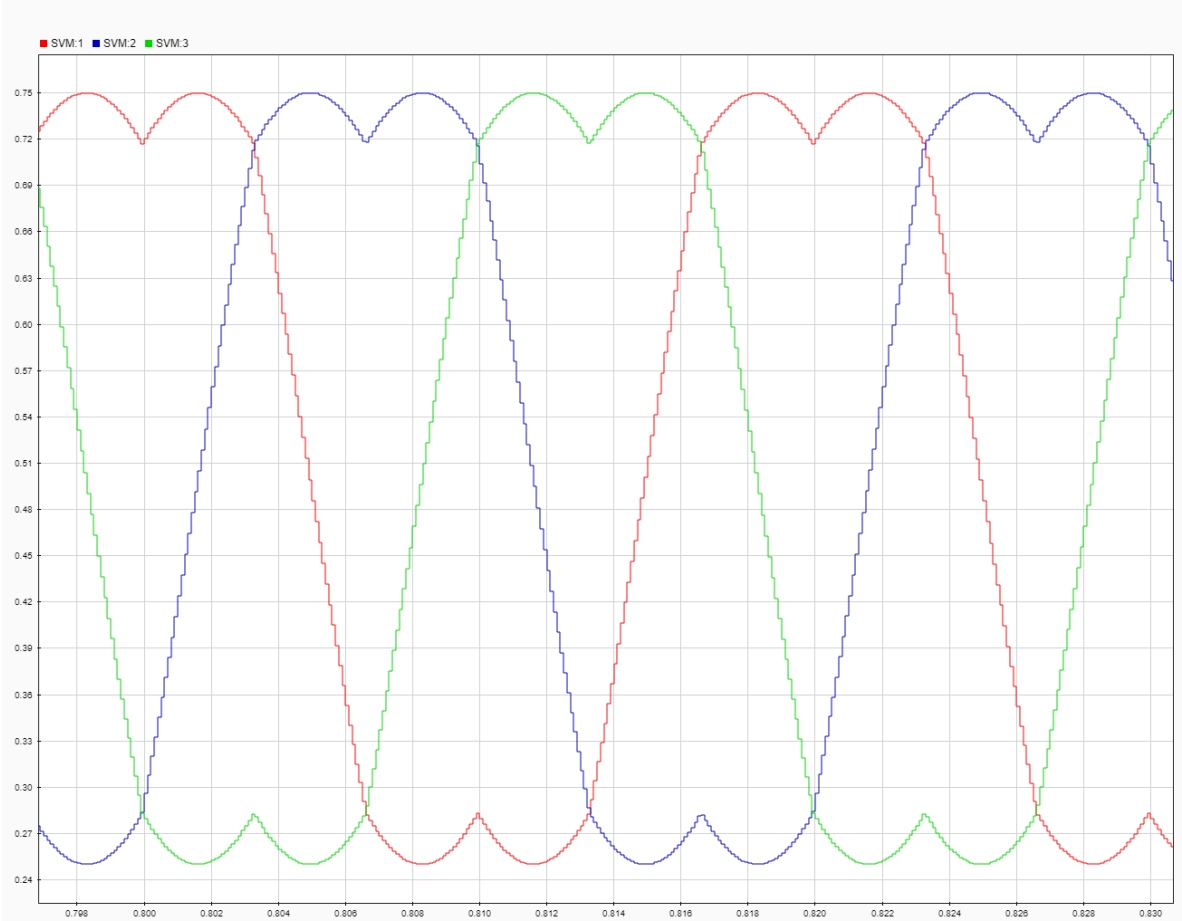

Figure 4. 15: Measured modulation signals from Simulink

This section provided an overview of the inverter and detailed its Simulink model. The following section will focus on the controller, providing an overview and validating the entire model.

4.1.3 Controller

Various control techniques exist for permanent magnet synchronous machines, including Field-Oriented Control (FOC), Direct Torque Control (DTC), and Predictive

Current Control (PCC). The models presented in this thesis are grounded in field orientated control (FOC).

Field oriented control or vector control is a commonly used control method for both induction machines (IM) and permanent magnet synchronous machine (PMSM) that makes full torque and wide speed range operation accessible. [44]

The implementation of field-oriented control ensures that the stator field direction remains orthogonal to the rotor field at all times. Consequently, the system response ripple is significantly less in comparison to six-step commutation, resulting in smooth machine operation. FOC is beneficial since it facilitates greater speed operation through the implementation of field weakening techniques. The advantages are accompanied by increased complexity. [44]

Zero torque is created when the rotor and stator magnetic fields are completely aligned, whereas maximum torque occurs when they are 90 degrees apart. FOC utilises the encoder's measured rotor angular position to calculate the required stator field vector, which is orthogonal to the rotor field by 90 degrees. The control algorithm additionally adjusts the current levels to attain the desired torque. The Clarke transformation is employed to convert the three-phase stator current into a two-phase axis (direct and quadrature-axis), wherein the q-axis stator current is accountable for torque creation, as elucidated in equation 3.19. Consequently, the d-axis stator current is maintained at zero while the q-axis stator current is optimised by the PI controller to get the requisite torque. Figure 4.16 illustrates the block diagram form of the Field-Oriented Control (FOC) algorithm employed in this project. [35, 44]

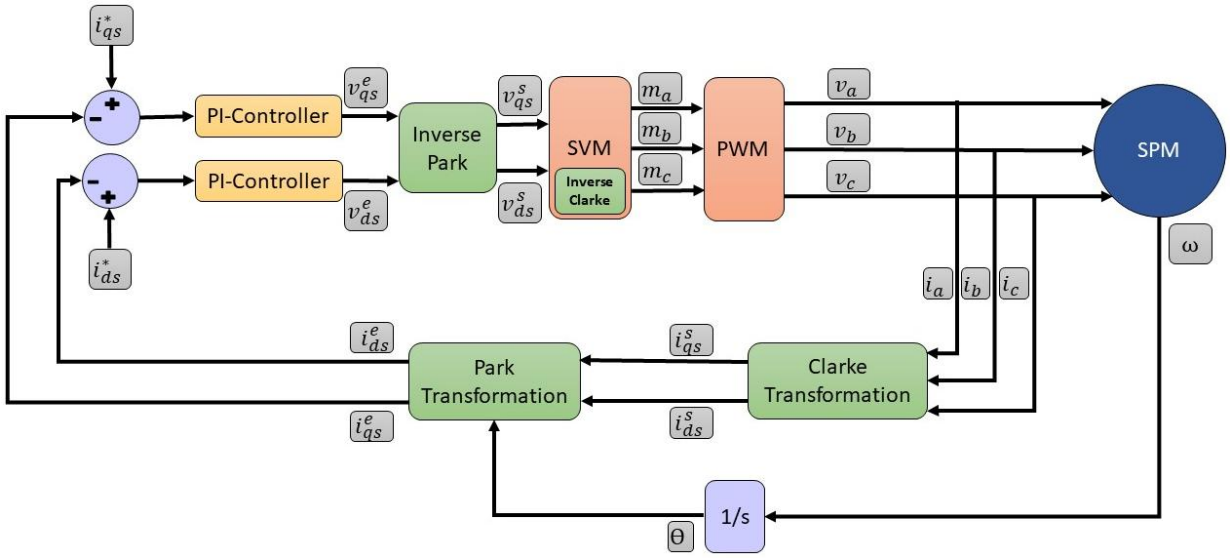

Figure 4. 16: FOC block diagram

The Ziegler-Nichols method has been employed to tune the PI controller. The tuning process involves incrementally raising the proportional gain Kp, while maintaining the integral gain at zero. The operation persisted until the output exhibited sustained oscillations. The Kp value at that juncture is designated as the crucial gain Ku. The oscillation period Tu was determined by monitoring the interval between consecutive oscillation peaks. The integral time constant Ti was calculated using equation (27). The computed Ti and Ku values were employed to determine the PI controller gains in accordance with the Ziegler-Nichols tuning approach, as specified in equations (26) and (28). Additional trial and error was conducted to attain optimal outcomes. [45]

$$K_p = 0.45\,.\,K_u \qquad (26)$$

$$T_i = \frac{T_u}{1.2} \qquad (27)$$

$$K_i = \frac{K_p}{T_i} \qquad (28)$$

To validate the model, it was operated in current control mode, demanding 50A, while operating at 1000rpm. Figure 4.17 depicts the phase current, whereas Figure 4.18 exhibits the observed dq-axis currents in proportion to their required values, confirming that the drive system is adequately stable and functioning efficiently. Figure 4.18 also illustrates the resultant torque from the model, which is approximately 2.4 Nm under the specified conditions.

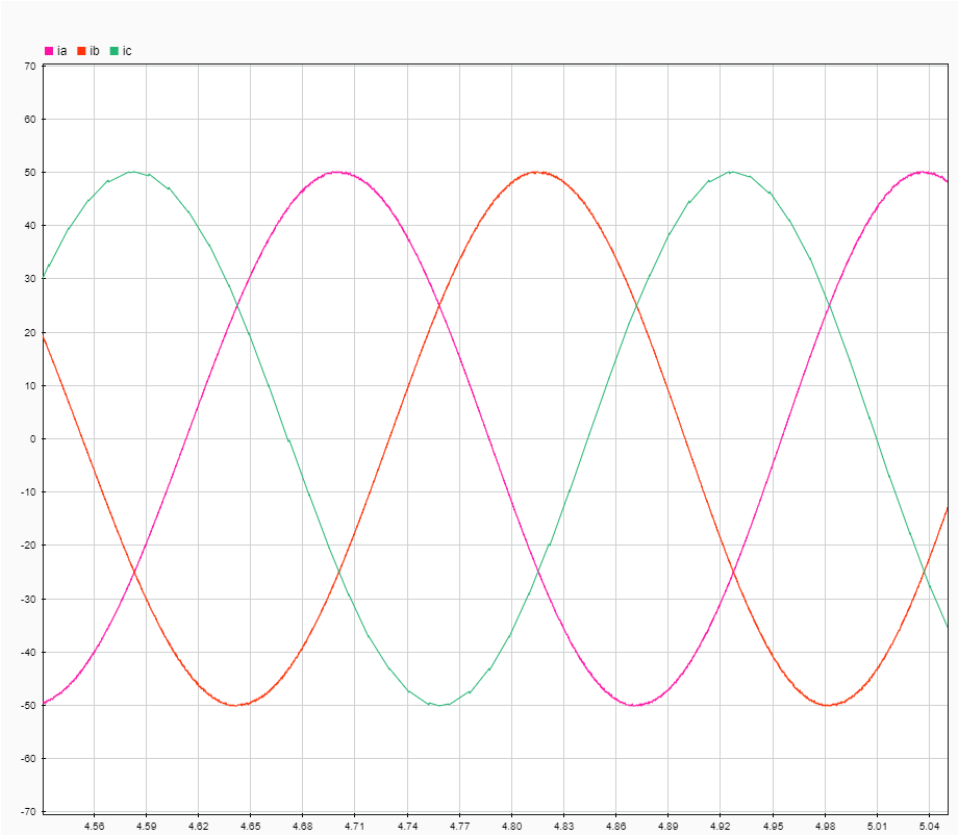

Figure 4. 17: Three-phase current Simulink model

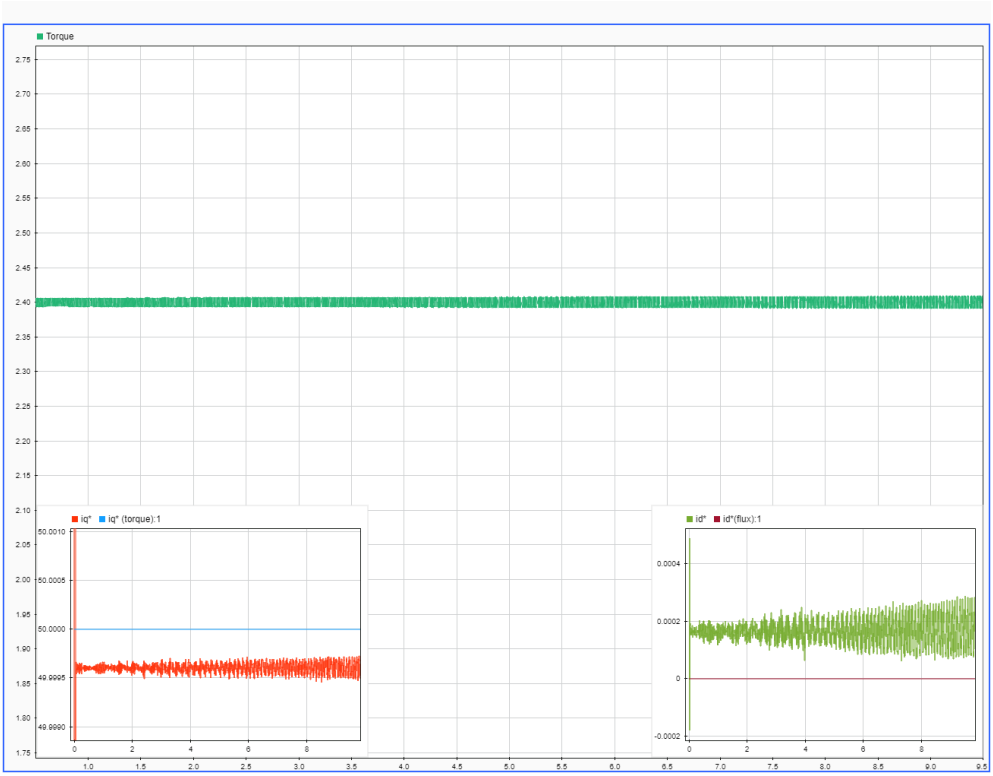

Figure 4. 18: Torque at 1000 rpm, zero D-Axis current and 50A Q-Axis current

In [12], Ari presents the torque values for various current levels in Figure 4.19. These results correspond to the machine utilised in this thesis. Based on Figure 4.19, the resultant torque from the generated model almost perfectly aligns with that of the actual machine.

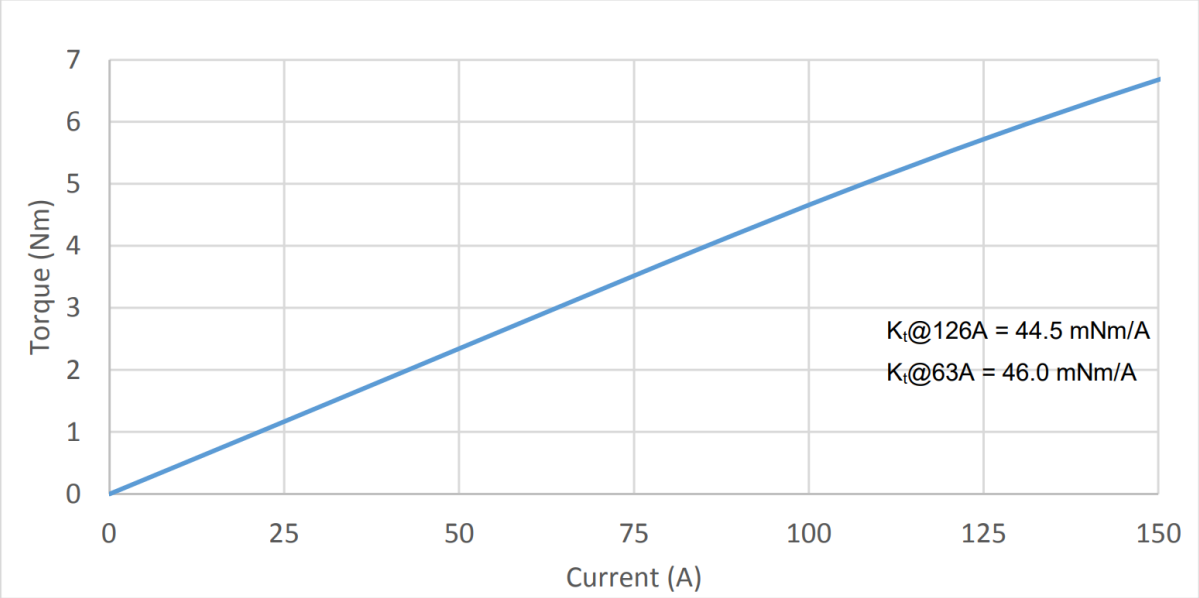

Figure 4. 19: Torque vs current based on 2D FE simulation in [13]

Figure 4.20 is derived from the data presented in Figure 4.19 and involves executing the model at various current levels. The discrepancy in torque between the model and the real hardware is determined to be 1.03%, which is rather accurate.

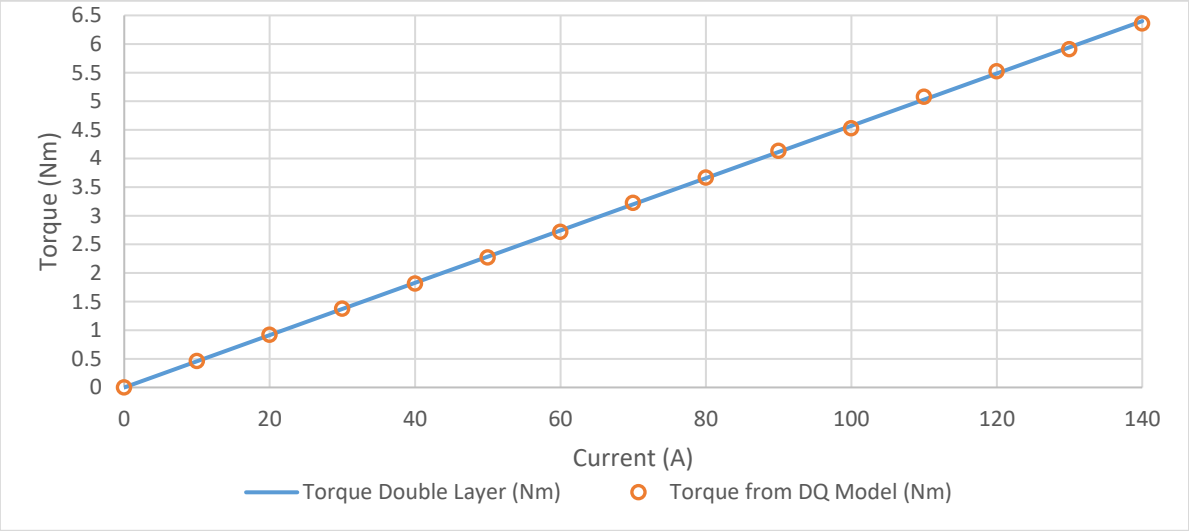

Figure 4. 20: Torque vs current comparison between double layer (motor used in this thesis) and the dq model at 1000 rpm

This section presents an overview of Field Orientated Control, effectively validating the model by comparing its results with actual motor outcomes retrieved from [13].The subsequent part will analyse the consequences of communication errors when various components of the model are integrated via a communication media, with a focus on the distributed model in loop simulation.

4.2 Summary:

This chapter delineates the three principal components of the SPM drive system: the SPM machine, the PWM and inverter, and the controller. This chapter emphasises the importance of utilising various Simulink solvers and their impact on the simulation's speed and accuracy. Furthermore, multiple stages of the modelling of the test rig in Simulink were elucidated, accompanied by a mathematical analysis.

The chapter also emphasised simulation outcomes, providing a comparison with actual hardware testing results that validated the model's efficacy. Figure 4.20 illustrates torque at various current levels, demonstrating the model's proximity to the actual drive system's performance. In conclusion, the first part of the project, illustrated in figure 4.21, has been given in this chapter, whereby a virtual model of the test rig has been built for further phases of the thesis.

Figure 4. 21: Hardware and virtual model components for the acquired test rig

The subsequent chapter concentrates on elucidating potential communication errors The subsequent chapter examines the effects of various communication issues, including packet loss, delay, and jitter, on an electric drive system. Results were exhibited on both the Simulink and dSpace Scalexio platforms.

Chapter 5

Communication Errors and System Performance Analysis

The preceding chapter was a crucial advancement in executing various Hardware-in-the-Loop (HiL) and Software-in-the-Loop (SiL) simulations. Now that we have constructed and approved the virtual model of the test rig, we can pursue the remaining steps of the project. This involves evaluating the effects of remote interconnection of several subsystems operating in real time as an integrated system.

This chapter presents a number of distributed X-in-the-Loop simulations that highlight the system's performance amid different communication challenges, such as packet loss and latency. The influence of communication faults might fluctuate based on the positioning of communication nodes within the simulation configuration. Therefore, this section evaluates three distinct configurations on both Simulink and dSpace platforms.

Figure 5.1 illustrates the initial configuration, which is the second phase of the research project proposal. In this configuration, the slave and master models are identical, and the slave model receives current feedback from the master instead of locally.

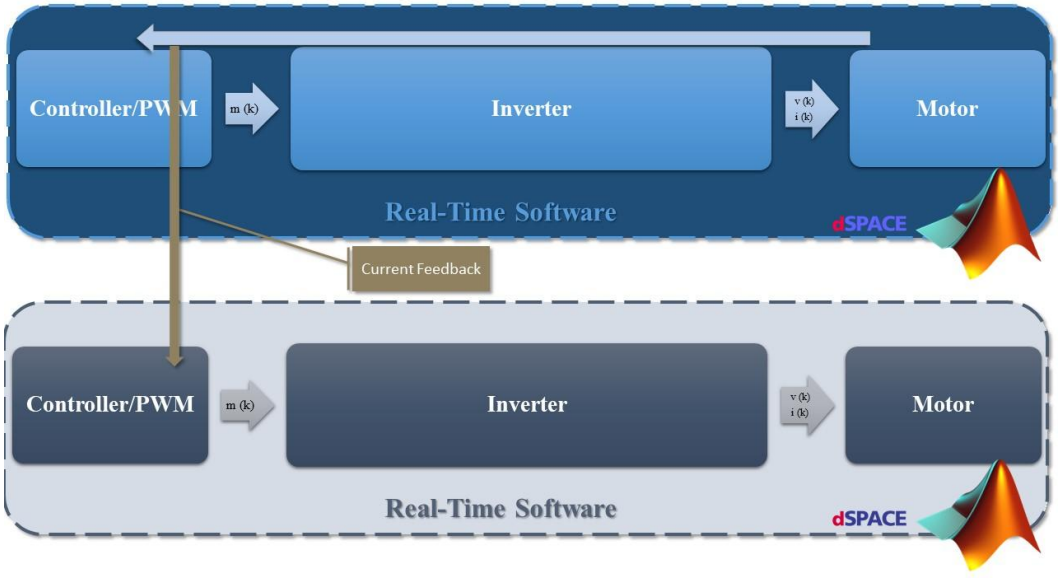

Figure 5. 1: The second phase of the project distributed real-time open-loop software-in-the-loop.

Figure 5.2 illustrates another arrangement in this work, where the slave model only consists of the electric model and receives voltage input from the master model.

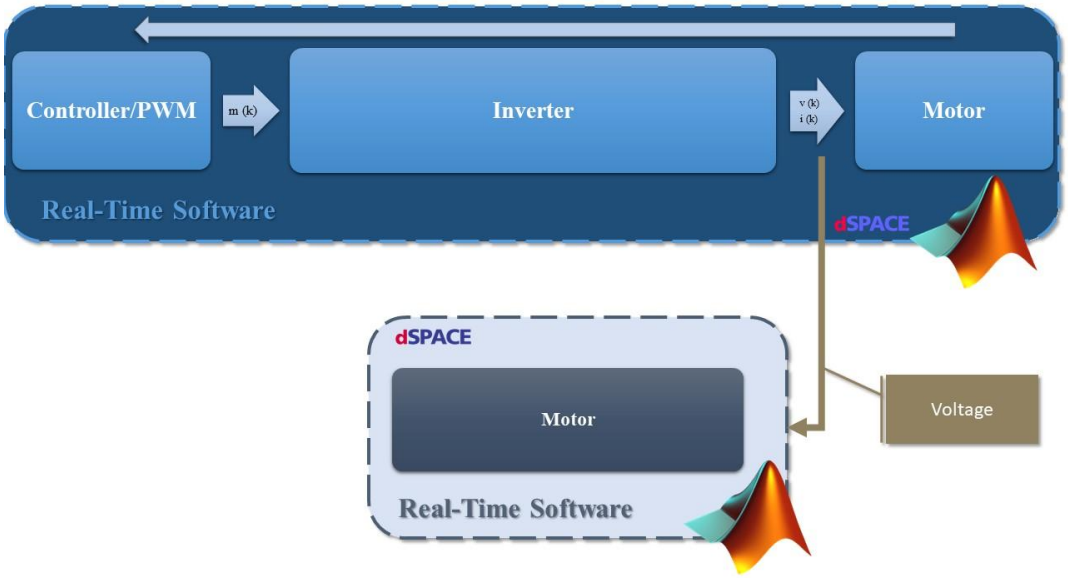

Figure 5. 2: The fourth phase of the project, distributed real-time open-loop software-in-the-loop.

The experimental configuration depicted in Figure 5.3 was executed and examined, in which the master system obtains torque feedback from the slave model, while the slave model functions depending on voltage inputs from the master model.

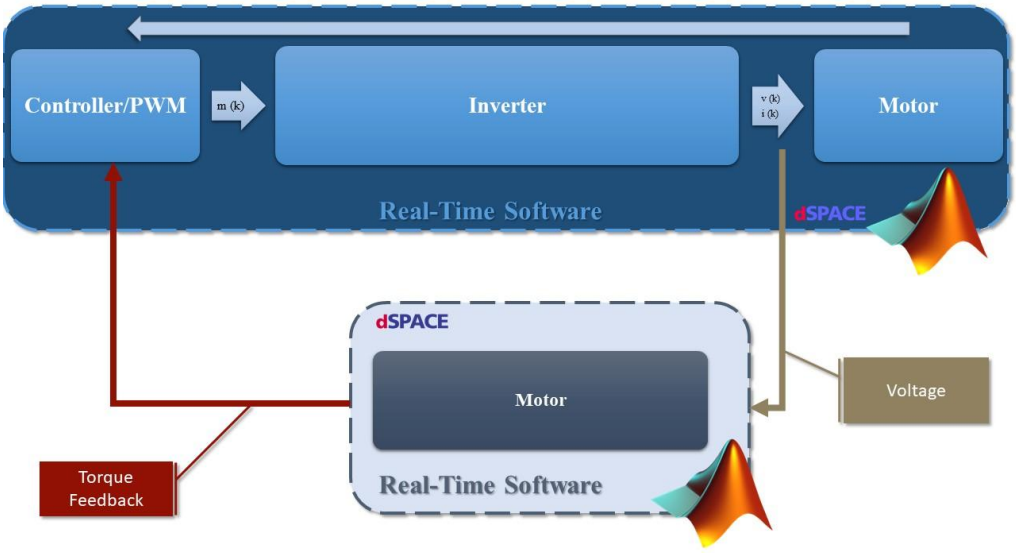

Figure 5. 3: Final stage of the project, distributed real-time hardware-in-the-loop

This series of tests has been executed subsequent to an examination of the network designated for this investigation. This review offers insight into the network and the likelihood of various issues arising inside it.

5.1 Network overview

This section examined the Janet network, the project's intended communication medium. Briefly elucidated in Chapter 1 that the JANET network, a high-capacity,

resilient infrastructure, serves the UK's research and education sectors. The network is responsible for handling approximately six million gigabytes of data on a daily basis. [12]

This section employs RIPE and PrefSONAR as tools for monitoring and measuring network performance over a period of 72hrs. RIPE facilitates policy formulation and resource administration for Internet operations across Europe, and PerfSonar, which is an open-source network measurement toolkit designed to monitor and diagnose network performance across multiple domains, is widely deployed in research and education networks to ensure optimal data transfer performance. [46, 47]

Despite Newcatsle being one of the nodes of the JANET network, the open-source tools used to analyse the network's performance lack sufficient data for this node. Therefore, we have considered Lancaster and London for further analysis. Throughput is the rate at which data is effectively transported from one location to another within a network in a designated timeframe, commonly quantified in bits per second (bps). Figure 5.4 shows that, despite the network being relatively stable in terms of throughput, it occasionally experiences unstable moments, Figure 5.4 is scaled to emphasis the alterations. Factors such as latency, packet loss, jitter, bandwidth, network congestion, and protocol overheads influence throughput. Original PrefSONAR screenshots can be found in Appendix 19. [48]

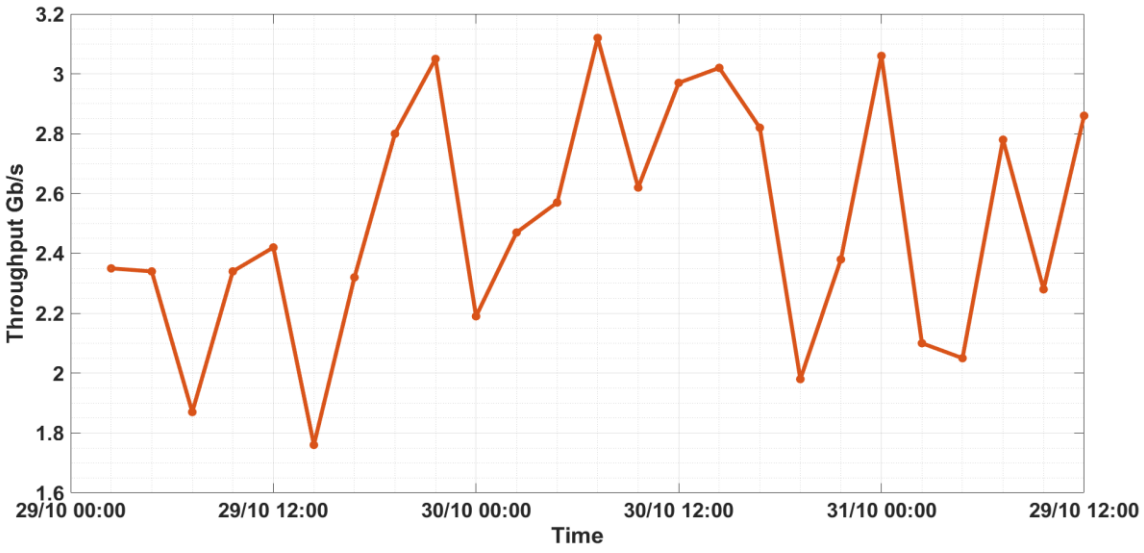

Figure 5. 4: PrefSONAR throughput analysis of JANET network [49]

This thesis employs Internet Protocol version 4 (IPv4), taking into account that the JANET network's backbone operates at speeds of up to 600 Gbit/s, thereby

significantly reducing network congestion. The primary factors under investigation are packet loss and latency. [12]

Packet loss occurs when data packets traversing a network fail to reach their designated destination. This issue can significantly impact the efficacy and reliability of network connections. Figure 5.5 shows packet loss measurement over 72hrs on JANET network. Despite the Janet network's design for outstanding dependability, even little packet loss can profoundly affect real-time Hardware-in-the-Loop (HIL) simulations. A packet loss rate of even 0.1% can diminish throughput by more than 50% in conventional TCP protocols, negatively impacting time-sensitive applications such as HIL simulations. [26, 29]

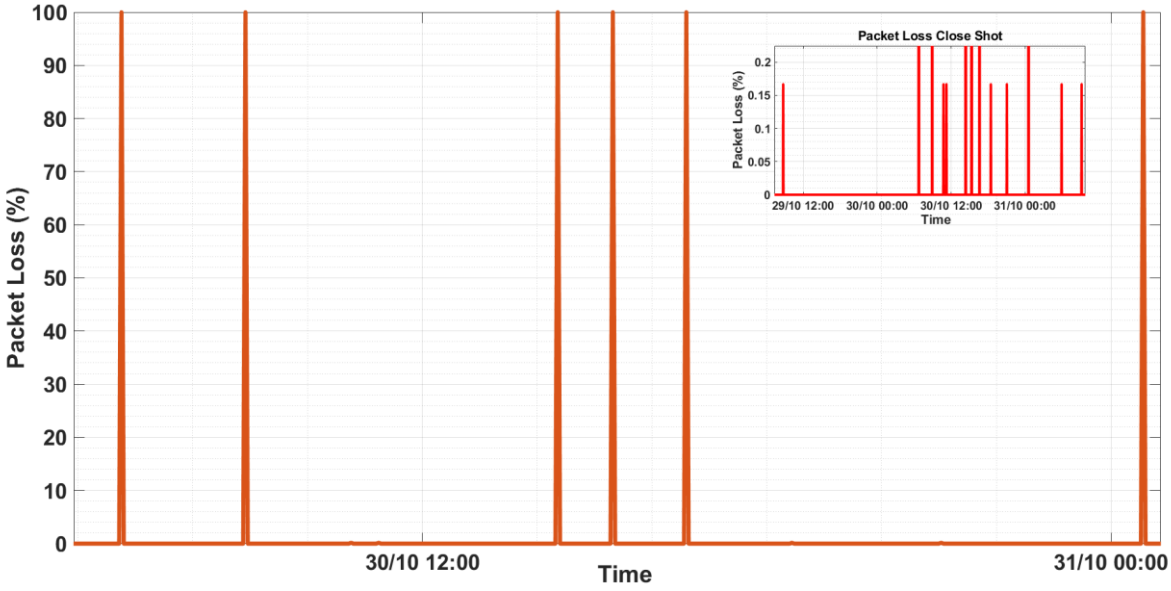

Figure 5. 5: PrefSONAR throughput analysis of JANET network [49]

To emphasise the impact of packet loss in an internet-distributed HiL simulation, equation (30) replicated the system using a discrete transfer function G(z). The probability of packet loss at each transmission interval is denoted as P_{loss}. [26, 29]

$$G_{loss}(z) = G(z).(1 - P_{loss}) \qquad \text{equation (30)}$$

Packet loss causes an unpredictable interruption in the update cycle, thus altering the system response, which may be further analysed by examining the system control error E(z), as described in equation (31), where H(z) signifies the feedback channel and R(z) symbolises the reference input.

$$E(z) = \frac{P_{loss}}{1 - G(z).H(z)} R(z) \qquad \text{equation (31)}$$

The multiplication term in equation (30), (1−Ploss), reduces the system's gain, indicating the average effect of absent updates, which undermines control accuracy. An increased Ploss results in a spike in error, hence destabilising the system.

Furthermore, another factor to consider is latency in the communication between two subsystems. Latency in Internet-Distributed Hardware-in-the-Loop (Internet HiL) for electric drive system induce desynchronisation between actual and virtual components, resulting in errors in the simulation outcomes.

The aforementioned analysis on the JANET network indicates that the delay is rather modest, as seen in Figure 5.6; yet, even little latency may significantly impact a dynamic hardware-in-the-loop simulation. The investigation indicated an average delay of 6ms between the two nodes. According to [50], data transmission delay may go up to 270ms at times. Therefore, it necessitates consideration in a distributed HiL simulation context.

Figure 5. 6: PrefSONAR latency analysis of JANET network [49]

Additionally, latency adds a delay τ that modifies the system's dynamic responsiveness. Latency results in a phase shift inside the control system. In an electric drive system characterised by a transfer function G(s), the introduction of a delay τ produces an alteration in the transfer function, equation (32) represents this behaviour.

$$G_{delayed}(s) = G(s).e^{-\tau s} \qquad \text{equation (32)}$$

The exponential component $e^{-\tau s}$ generates a phase shift that destabilise the system if τ is significant compared to the system's natural frequency. therefore, it is essential to address latency in any dynamic system.

Furthermore, Jitter is a characteristic of a communication network that denotes the variability in packet arrival timings, usually defined as the standard deviation or range of latency, measured in milliseconds (ms).

A recognised delay in a HiL simulation could be mitigated using time-shifting or predictive compensation techniques, while jitter requires more complex compensation measures. The JANET network, as previously described, has an average jitter of 0.281ms between the nodes, depicted in Figure 5.10. [49]

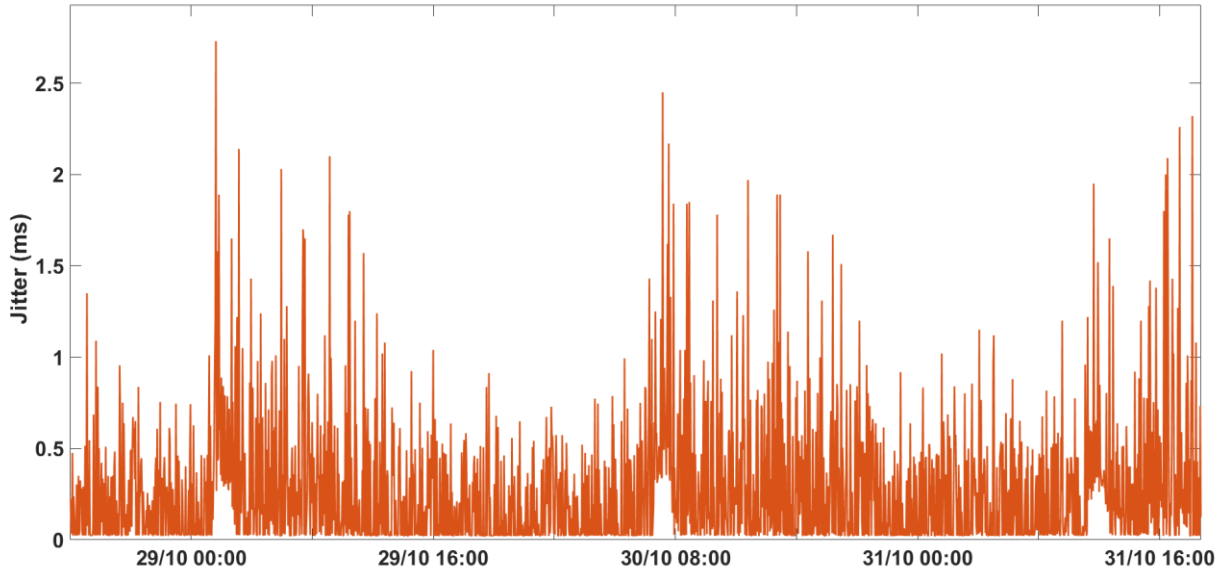

Figure 5. 7: PrefSONAR jitter analysis of JANET network [49]

Incorporating a term, J, into equation (32) induces stochastically varying phase shifts, resulting in unexpected timing of the system's response. Equation (33) indicates that significant jitter in a highly dynamic system, such as an electric drive system, would substantially affect its stability. [51]

$$G_{delayed}(s) = G(s).e^{-(\tau+J(s))s} \qquad \text{equation (33)}$$

Despite the low average jitter on the JANET network, it experiences spikes of up to 2.5 ms, which will dramatically increase when considering nodes that are geographically distant from one another. Consequently, jitter as an important factor in communication must be handled when subsystems are virtually interconnected in remote places.

This section emphasised several aspects of the communication network inside this thesis. The following section examines the effects of packet loss, delay, and jitter on the simulations of electric drive systems across various Xil environments.

5.2 Impact of Communication Issues on Electric Drive System Performance

This section explores the effects of difficulties with communication across three distinct configurations shown in Figures 5.1, 5.2, and 5.3. The study is divided into three subsections, each corresponding to one of the XiL configurations. Moreover, packet loss is replicated by a standard model from MathWorks, depicted in Appendix 9, that operates SimEvents® Entity Output Switch block. In addition, Delay blocks have been used to simulate latency in the system. Appendix 22 illustrates the packet loss blocks, as well as the time delay blocks, and the jitter blocks employed in this section of the thesis. [52]

5.2.1 Influence of network impairments on the second phase of the project

This section examines the configuration illustrated in Figure 5.1, whereby both the master and slave models contain an entire SPM machine driving system. The slave and master models are executed on two distinct processing units of the dSpace Scalexio. The slave model receives the three-phase current feedback from the master model. The slave model lacks a feedback loop, and the torque generated by the slave is relying on the current feedback signals from the master. Additionally, In this section, the electric drive system was operated with 2.4Nm of load and at a constant speed of 1020rpm.

Moreover, as stated in the preceding section, three network impairments will be examined in relation to the electric machine drive system developed earlier in this part. comprising outcomes from both Simulink and dSpace Control Desk. This comprehensive approach will yield a profound understanding of how various limitations influence overall system functionality.

This section details the attempt to use the test rig's current feedback signal to operate the Simulink model of the rig as a subordinate model, demonstrating hardware-in-the-loop simulation, the setup is depicted in figure 5.11. The current clamps monitored the three-phase current signal and transmitted it to the dSpace Scalexio, allowing the Simulink model to acquire the test rig's current data.

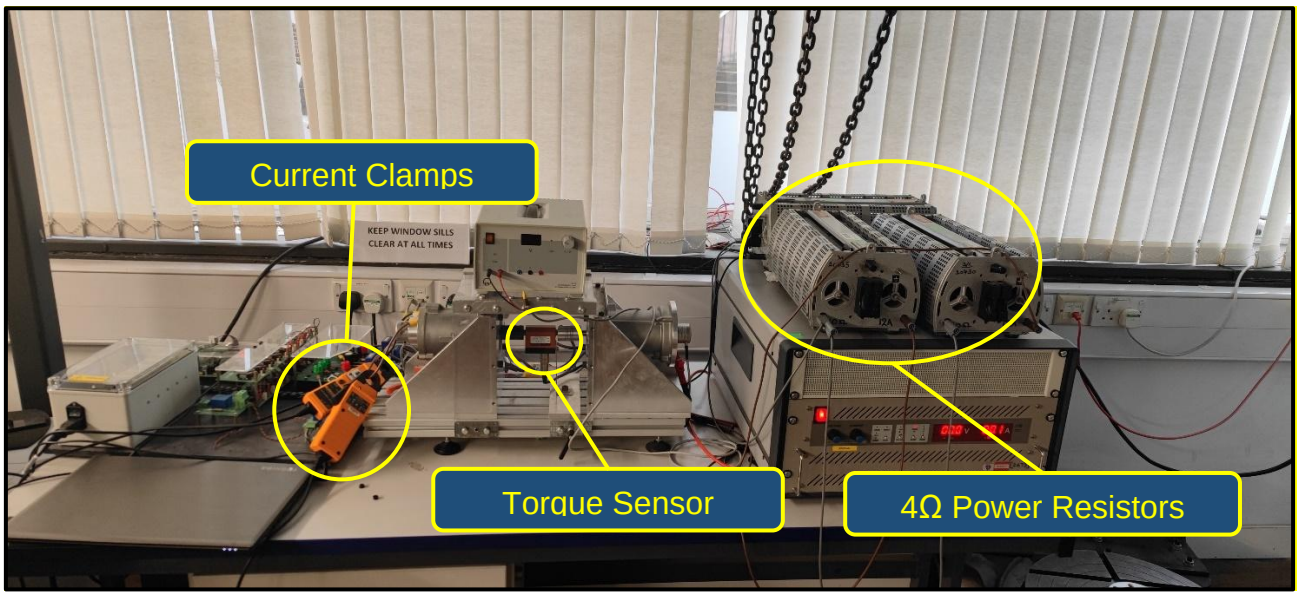

Figure 5. 8: Test Rig with Power Resistors Connected

The current clamps (shown in Appendix 7) used for current signal measurement do not deliver a clean signal. Hence, active low-pass filters were used to attenuate the noise in the signal. For precise simulation, the Simulink model required amplification of the signal from the current clamp. Figure 5.12 illustrates the procedures used to provide current from the test equipment to the dSpace Scalexio.

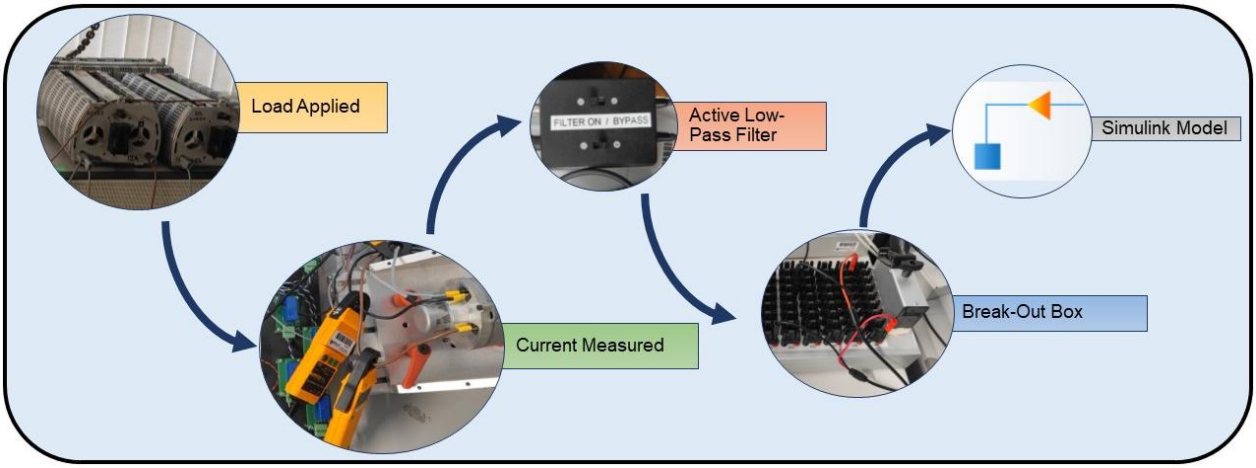

Figure 5. 9: Test Rig Current to Simulink Steps

In a field-oriented control system, packet loss in a three-phase current feedback signal can result in missing data points and erroneous or incomplete feedback to the controller. This significant compromise in system performance could potentially lead to instability, shutdown, and potential harm to the system.

The torque equation in the dq reference frame of an SPM machine is represented by equation (34). T denotes torque, while ψ signifies flux linkage. Packet loss will lead to

errors in the iq, causing the controller to struggle with stability. This will disrupt accurate torque production and significantly impact performance, especially during transients.

$$T = \frac{3}{2}\frac{P}{2}\psi_m i_q^r \qquad (34)$$

Figure 5.13 illustrates system performance during instances of packet loss in the Simulink model. Figure 5.14 similarly illustrates a packet loss scenario in the Control Desk. Both tests emphasize the significance of packet loss and its critical impact on system stability and performance.

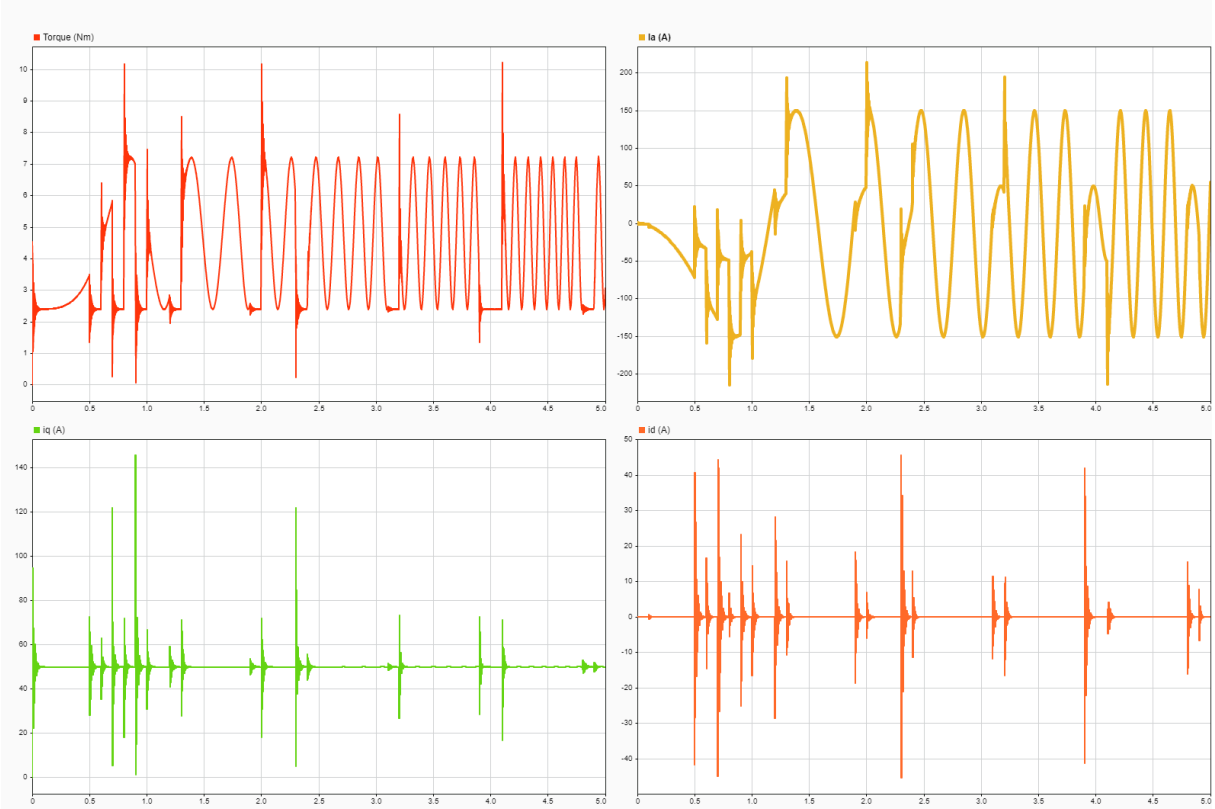

Figure 5. 10: Second phase packet loss test (30%) - Simulink

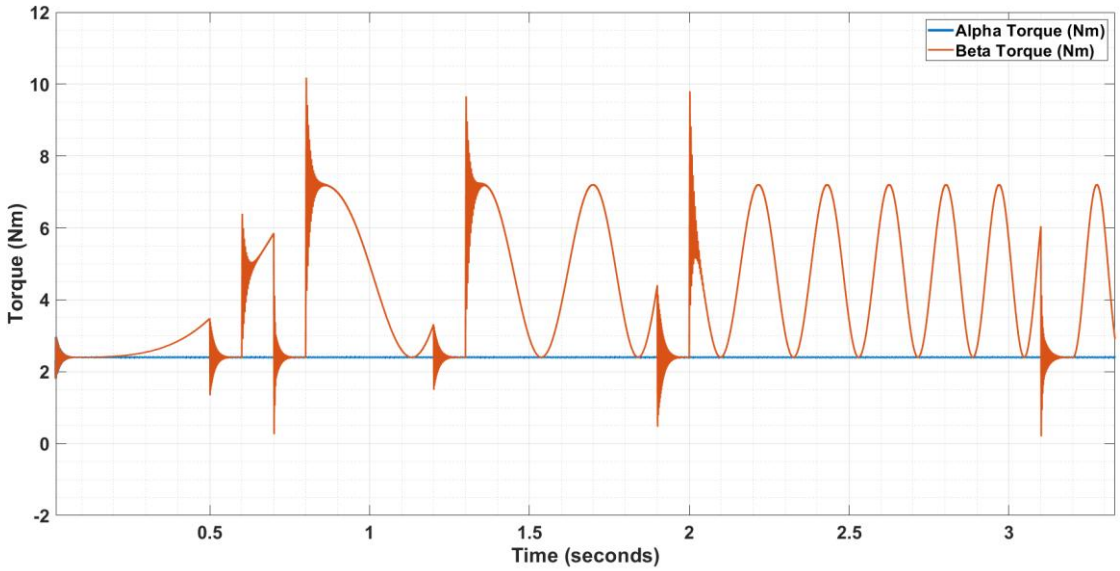

Figure 5. 11: Second phase packet loss test (20%) - dSpace ControlDesk

Another element to consider is latency, which may potentially affect the performance and stability of the drive system. In a field-oriented control system, latency in feedback signal directly affects torque and flux control, resulting in errors and discontinuities. Analysing the voltage equation in the dq reference frame, equation (35) and equation (36), considering that the current components experience a time delay of τ. Therefore, the real-time I_{dq} (t) will change into i_{dq} (t+τ), that leads to asynchrony in the control system. This discrepancy between flux and torque results in instability and poor performance.

$$v_d^r = r_s i_d^r + L_s \frac{d}{dt} i_d^r - \omega L_s i_q^r \qquad (35)$$

$$v_q^r = r_s i_q^r + L_s \frac{d}{dt} i_q^r + \omega L_s i_d^r \qquad (36)$$

To demonstrate this point, latency has been modelled and evaluated using both Simulink, seen in Figure 5.15, and dSpace Scalexio, shown in Figure 5.16. The tests indicate that the controller's reaction to voltage instructions is delayed, resulting in oscillation in the torque and causing instability.

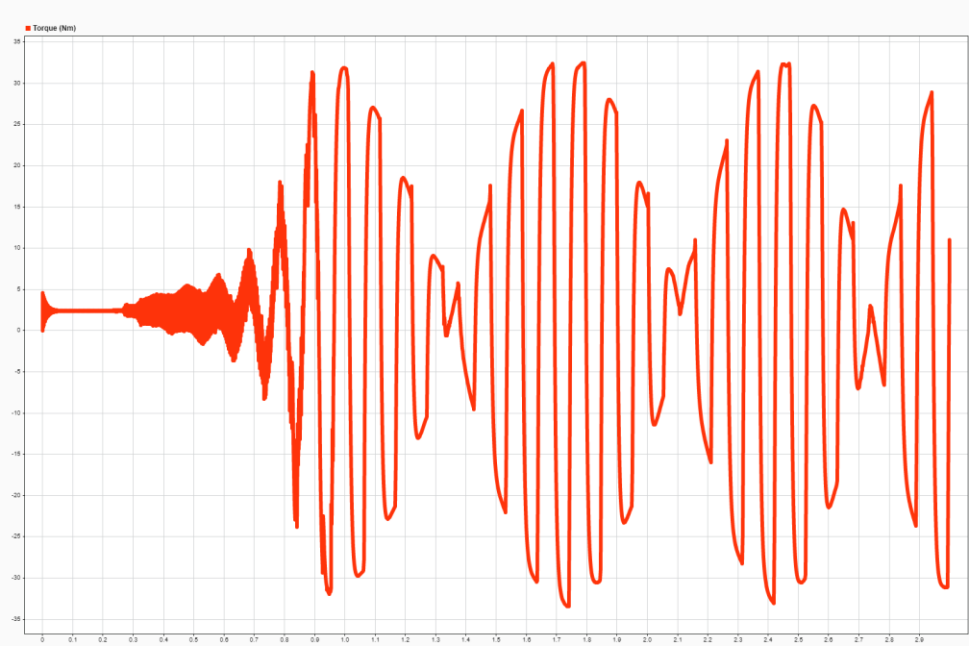

Figure 5. 12: Second phase latency test (20ms) – Simulink

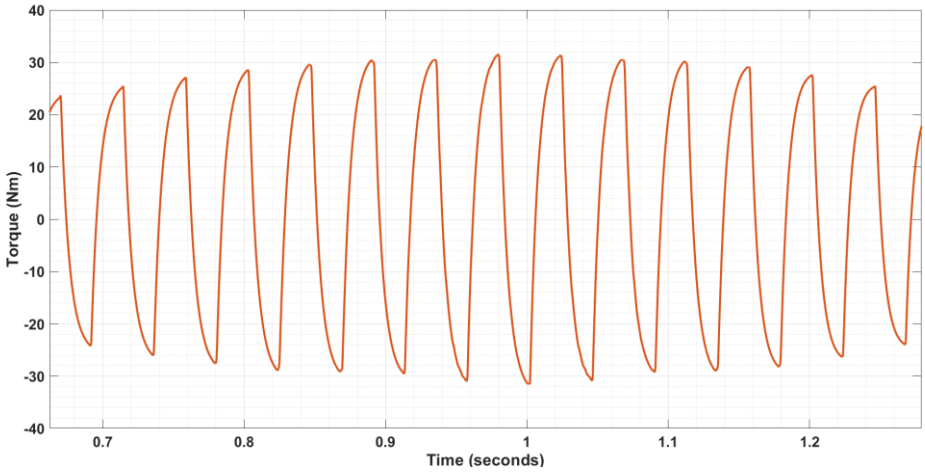

Figure 5. 13: Second phase latency test (50ms) – dSpace ControlDesk

This stage of the thesis also examines jitter, which is the inconsistency of latency in packet delivery. Timing inconsistency in electric drive system's current feedback leads to incorrect current sampling, resulting in a mismatch between the expected and actual current feedback. This mismatch disrupts the alignment of the stator and the rotor magnetic fields.

Referring to equation (37), jitter has the potential to cause the idq values to lose their accuracy at the required control time. This, in turn, can affect the torque demand, leading to excessive demand and system instability. This is clear from both Simulink and dSpace test results depicted in Figure 5.17 and Figure 5.18.

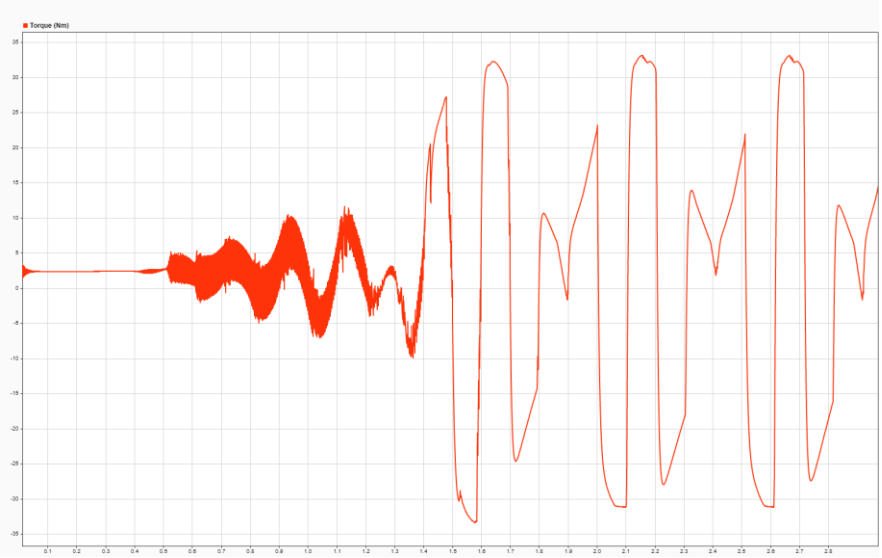

Figure 5. 14: Second phase Jitter test – Simulink

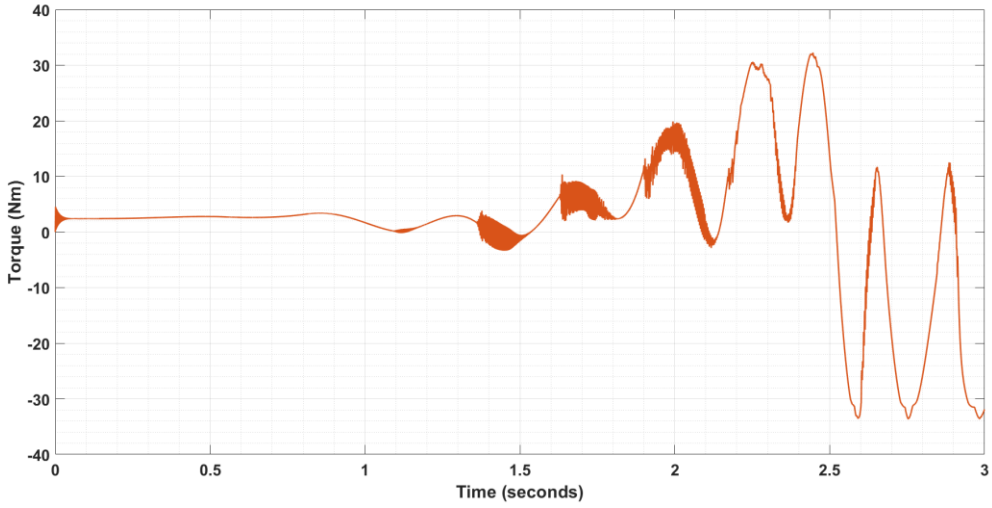

Figure 5. 15: Second phase Jitter test – dSpace ControlDesk

This section has examined the impact of jitter, latency and packet loss in the second phase of the thesis. The next section highlights the influence of these network impairments in the third phase of the thesis, depicted in figure 5.2.

5.2.2 Impact of network impairments on the third phase of the project

Figure 5.2 illustrates the third phase of this research project, where the slave model solely comprises the electric machine model, with the voltage signal serving as the input signal from the master model. Similar to the previous section, in three stages examine the impact of network issues on this configuration, highlighting the effects of each element: packet loss, jitter, and latency.

In the arrangement shown in figure 5.2, similar to the second phase, packet loss is impairing the overall performance of the system. The communication nodes in this

section differ, and the signal in question is a three-phase voltage signal from the inverter, characterized by a square waveform with a frequency of 16 kHz, corresponding to a period of 62.5 µs. This frequency is considerably higher than the current signal previously discussed, which had a frequency of 68 Hz at 1020 rpm.

Packet loss in voltage signals impacts the accurate execution of dq voltages, resulting in erroneous decoupling of d-q axis currents, apparent in equation (35) and equation (36), which leads to torque fluctuation and control instability. Hence, it substantially diminishes the performance of the SPM drive system by decreasing control precision.

The Simulink results in Figure 5.19 and the ControlDesk results in Figure 5.20 show how packet loss affects the performance of the machine drive system compared to when the drive system is working normally (which we talked about in Chapter 4).

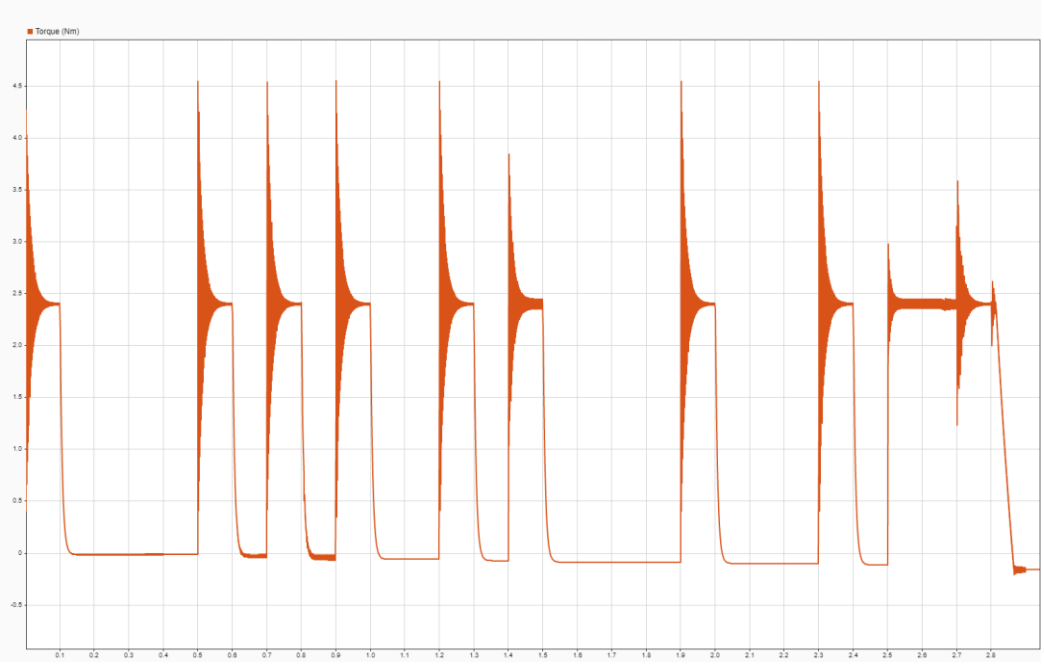

Figure 5. 16: Third phase packet loss test (30%) - Simulink

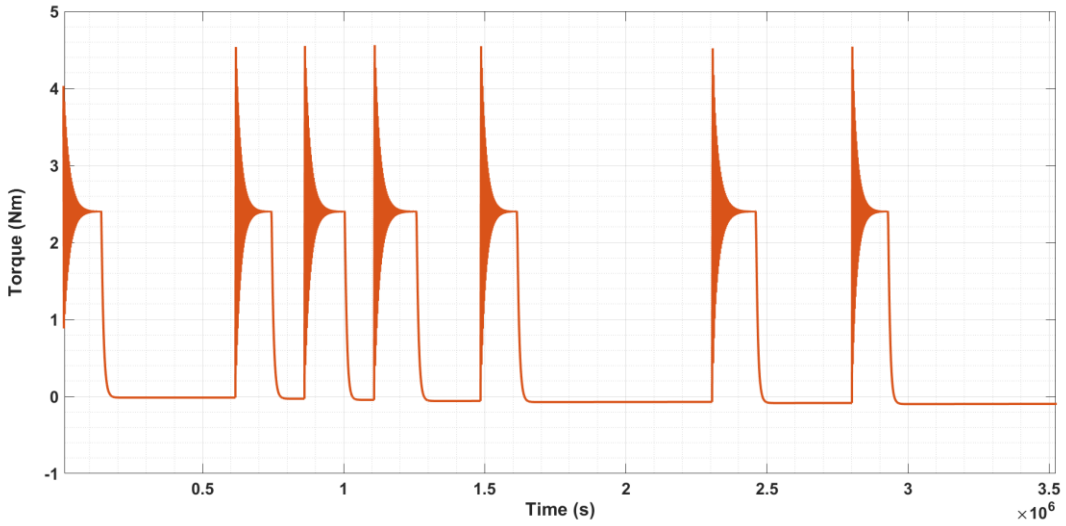

Figure 5. 17: Third phase packet loss test – ControlDesk

This section describes a slave paradigm that lacks a feedback signal. Consequently, the latency in the voltage signals will result in a delay in torque delivery in the slave model; thus, the produced torque will exhibit a delay equivalent to τ, this has been advanced from the voltage signals as shown in equation (37) and equation (38). This has been illustrated in Figure 5.21.

$$v_d^r(t) = r_s i_d^r(t-\tau) + L_s \frac{d}{dt} i_d^r(t-\tau) - \omega L_s i_q^r(t-\tau) \quad (37)$$

$$v_q^r(t) = r_s i_q^r(t-\tau) + L_s \frac{d}{dt} i_q^r(t-\tau) + \omega L_s i_d^r(t-\tau) \quad (38)$$

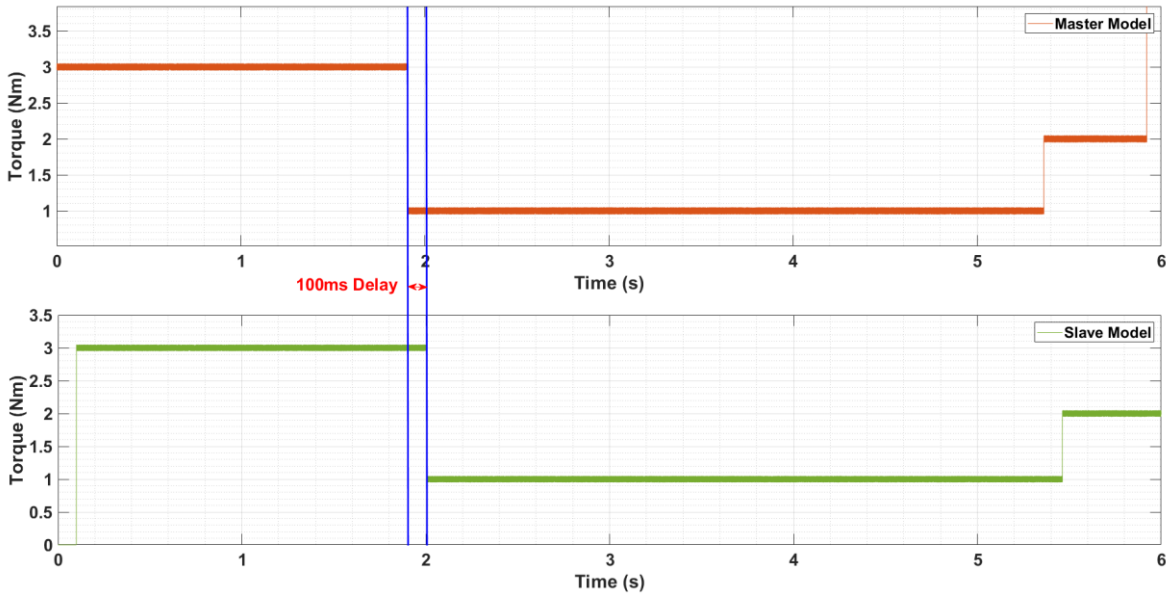

Figure 5. 18: Third phase latency test (100ms) – dSpace Control Desk

When the system experiences jitter, the slave model will exhibit a variable time delay; thus, the torque produced by the slave model incorporates a J_τ, random latency, component in the dq voltage equation (39) and equation (40), emphasizing the system's changing latency, which impacts torque output and the system's stability. [51]

$$v_d^r(t) = r_s i_d^r(t - J\tau) + L_s \frac{d}{dt} i_d^r(t - J\tau) - \omega L_s i_q^r(t - J\tau) \quad (39)$$

$$v_q^r(t) = r_s i_q^r(t - J\tau) + L_s \frac{d}{dt} i_q^r(t - J\tau) + \omega L_s i_d^r(t - J\tau) \quad (40)$$

Figure 5.22 illustrates the situation when jitter occurs in the system, as highlighted in the figure. When there is a change in latency, the slave mode may not have corresponding values from the master model due to insufficient data, which can impact the system in a manner similar to packet loss.

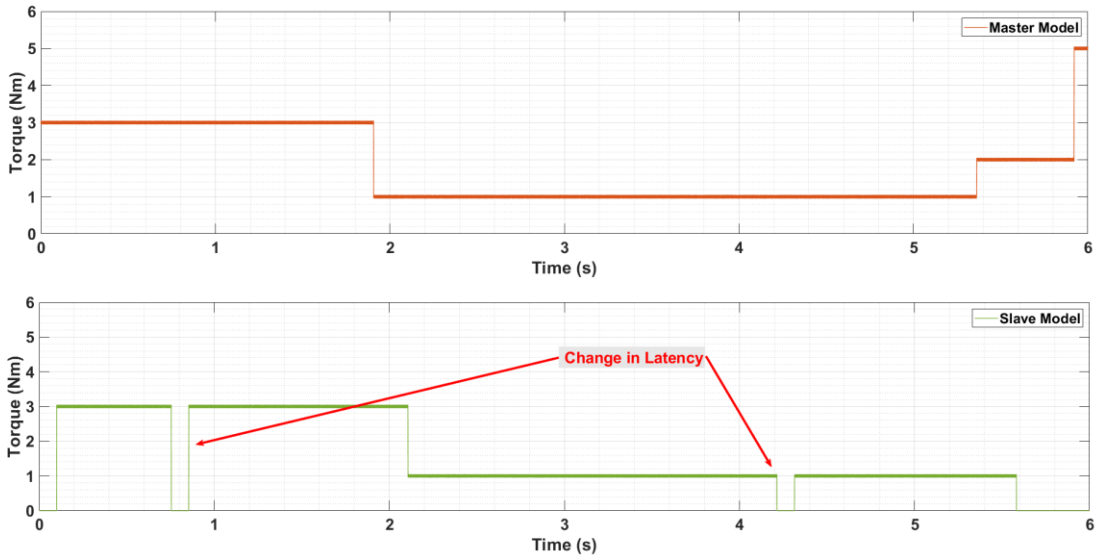

Figure 5. 19: Third phase jitter test – dSpace Control Desk

5.2.3 Impact of network issues on the fourth phase of the project

Figure 5.3 illustrates the fourth phase of the project, which represents the ultimate goal of the thesis. Unlike phases two and three, this stage incorporates a remote feedback loop, which heightens the complexity of the simulation by allowing network impairments to affect both the master and slave models. Unlike the previous two sections, this section considers both master and slave model results.

Packet loss in the slave model input will also impact the performance of the master model through the feedback signal. Considering a feedback torque value of zero will results in maximum torque demand while this feedback value is only true for the slave model. Figure 5.23 illustrates the torque signals of the slave and master in a packet loss situation.

In this circumstance, the master model controller does not get a feedback signal; hence, it will inaccurately adjust the q-axis current. On the other hand, in the slave model, during a packet loss scenario, the motor will receive no power signals, leading to a torque output of zero in the model. At the software level, these situations are not destructive; however, in real-world scenarios, the system may undergo a safety shutdown or perhaps incur hardware damage.

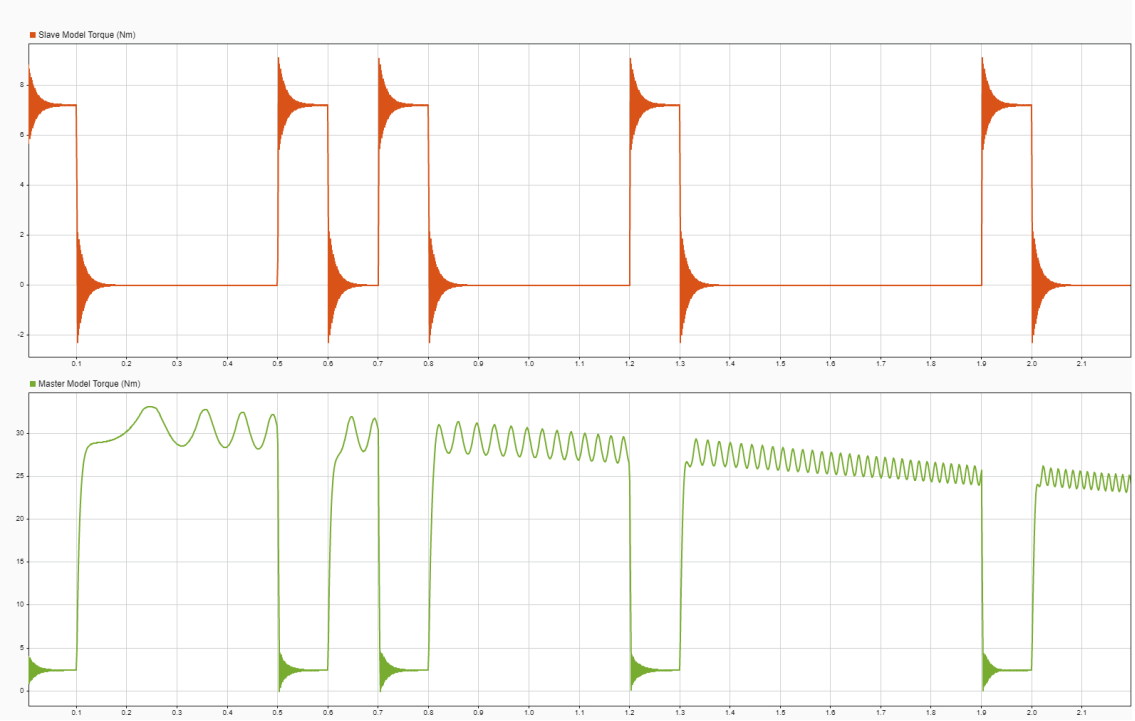

Figure 5. 20: Fourth phase packet loss test - Simulink

In the event of jitter and latency, they cause instability in the real-time simulation due to phase mismatch, phase difference between the reference and feedback, and

delayed response due to feedback delay. Figure 5.24 illustrates the master model torque signal that has been heavily impacted by the asynchronicity in the system.

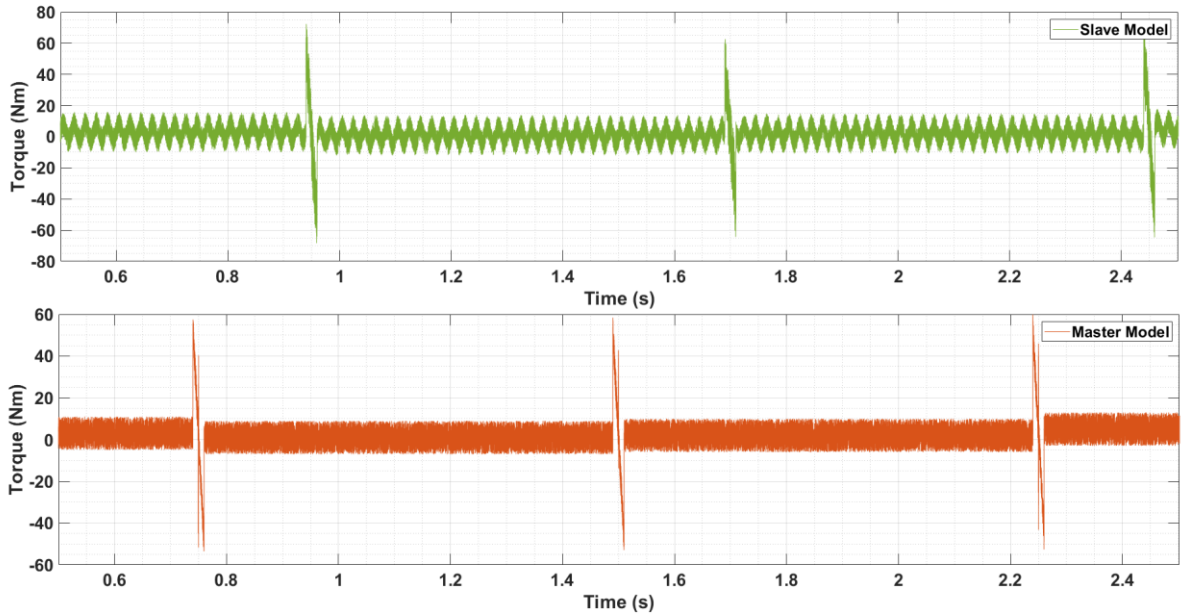

Figure 5. 21: Fourth phase Latency test – ControlDesk

Consider the SPM machine torque equation (37), which illustrates the addition of a latency term, τ, to the calculation of torque output at any given time.

$$T(t) = \frac{3}{2}\frac{P}{2}\psi_m i_q^r(t-\tau) \qquad (34)$$

Similar to the jitter test result from the previous section, in case of network experiencing jitter, a change in time delay may result in the slave mode missing packets from the master model due to inadequate data, thereby affecting the system comparable to packet loss. this has been depicted in Figure 5.25.

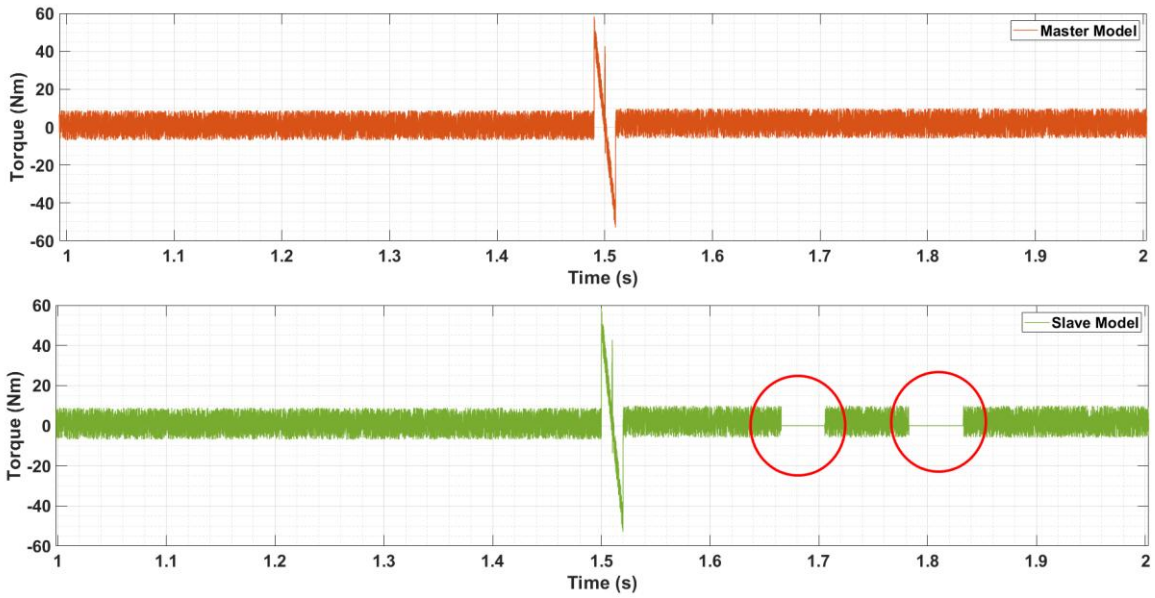

Figure 5. 22: Fourth phase jitter test – ControlDesk

5.3 Summary

The Simulink model developed in chapter four was utilized alongside the implication of the same model on the dSpace to investigate the impact of network impairments, including latency, jitter, and packet loss. Changes in the communication node and overall simulation setup affect the system differently, but they nonetheless cause instability in different magnitudes and significantly impact performance.

The effect of the network issue on the system can be summarized as follows:

- Packet loss
  - Results in missing feedback packet leads to control instability.
  - Leads to uncontrollable motor torque, vibrations, and overheating.
  - Causes emergency system shut down and hardware damage.
- Latency
  - Results in losing synchronization between control and motor response.
  - Delays in torque/flux adjustments leads to hardware damage through thermal stress and demagnetization of permanent magnets.
  - Results in significant overshoot and instability, especially during transient conditions.
- Jitter
  - Results in severe torque pulsations, destabilizing the system operation.
  - Induces excessive stress on mechanical components (in case of utilizing HiL), shortening their lifespan.

- Hardware damage due to high variability in data timing that leads to resonances in motor dynamics.
- In high-speed testing, jitter can lead to catastrophic system failures.

The following chapter explores the novel methodology of this thesis in tackling various network difficulties and presents findings to demonstrate its efficacy.

Chapter 6

Presenting a Novel Solution to Mitigate Network Impairment in Electric Drive Systems

Investigating the effects of packet loss, latency, and jitter on the SPM machine drive system in the previous chapter made it clear that a sophisticated approach is necessary to implement a distributed version of HiL and achieve a robust, reliable, and accurate setup.

Given that the models implemented on the dSpace Scalexio and Matlab are identical, and previous chapters have validated the dSpace simulations, this section solely presents results from the dSpace ControlDesk. Moreover, the real-time simulations at a high sampling rate of 10 microseconds need a highly capable processing unit, which is the reason for our acquisition of dSpace.

This chapter first tackles the resolution of packet loss using the second phase of the project that is an open-loop setup, depicted in Figure 6.1. Additionally, this chapter incorporates results from ControlDesk to validate the principles of the proposed novel approach using the test outcomes from the project's final phase, Figure 6.2, which constitutes the primary objective of this thesis. This section considered all three aspects of network impairment, including packet loss, latency, and jitter between the slave and master models, and validated the proposed approach.

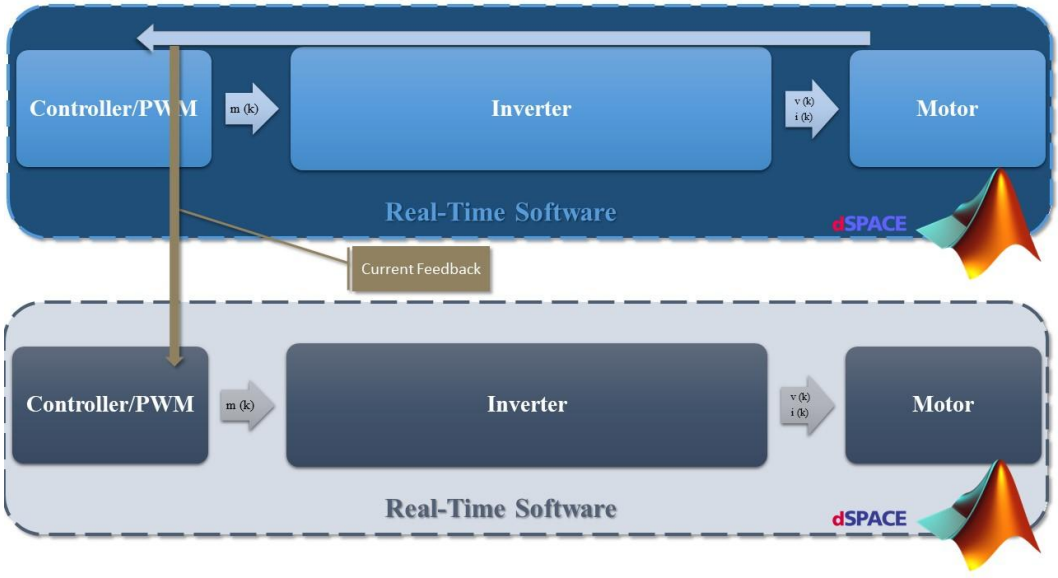

Figure 6. 1: The second phase of the project distributed real-time open-loop software-in-the-loop

Figure 6. 2: Final stage of the project, distributed real-time hardware-in-the-loop

6.1 Open-Loop Setup: EM Drive System Model in The Loop

Chapter 2 highlights the various research studies conducted on this topic, leading to the development of a variety of techniques to resolve packet loss. Some of the most common methods include leveraging artificial intelligence and machine learning [53], as well as developing a lookup table [54]. Another solution is to use other network protocols, such as Transmission Control Protocol, which is more robust than User Datagram Protocol and ensures the delivery of each packet. However, it is approximately 50% slower.

In this section, a communication architecture is developed based on previous research, specifically [27], and drive system requirements. This architecture is shown in Figure 6.3 and can be used with either a look-up table or an emulated slave model.

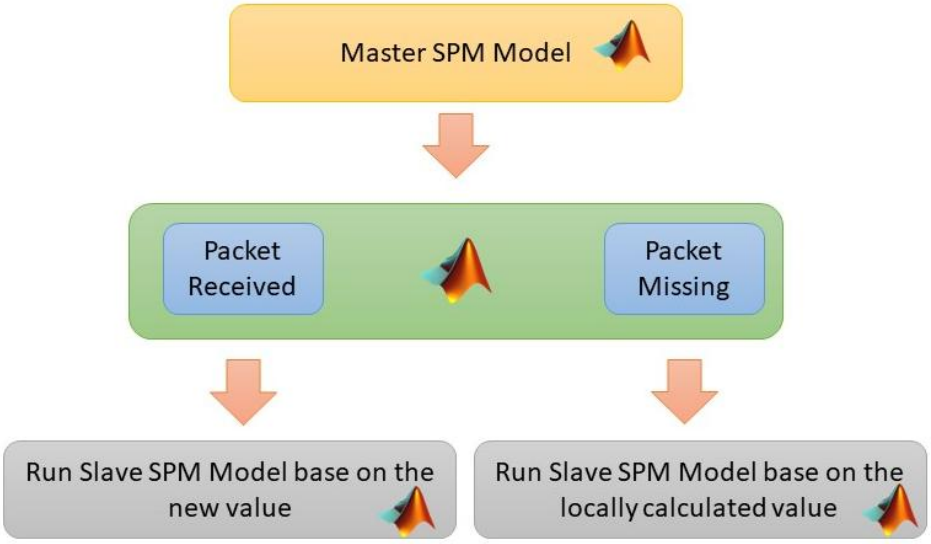

Figure 6. 3: Developed communication architecture flow-chart

Moreover, the slave and master models used in this section are identical, rendering the modelling mistake insignificant. An alternative solution is the creation of a lookup table. The suggested flowchart was executed in the subordinate Simulink model, as seen in Figure 6.4. Figure 6.5 illustrates the placement of the logic gates inside the slave Simulink model. The slave controller utilises the processed three-phase current signal obtained from the master model. When a packet is lost during communication, it will be substituted by a locally computed current feedback packet.

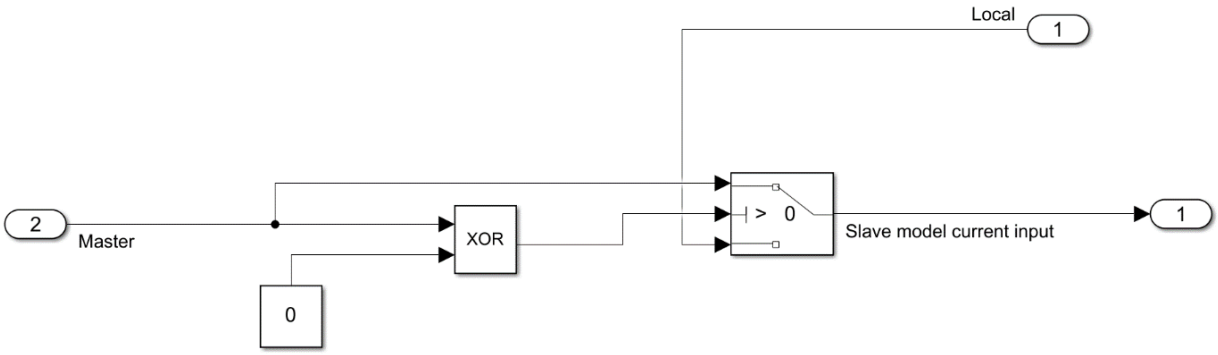

Figure 6. 4: Logic gates to implement communication flow chart.

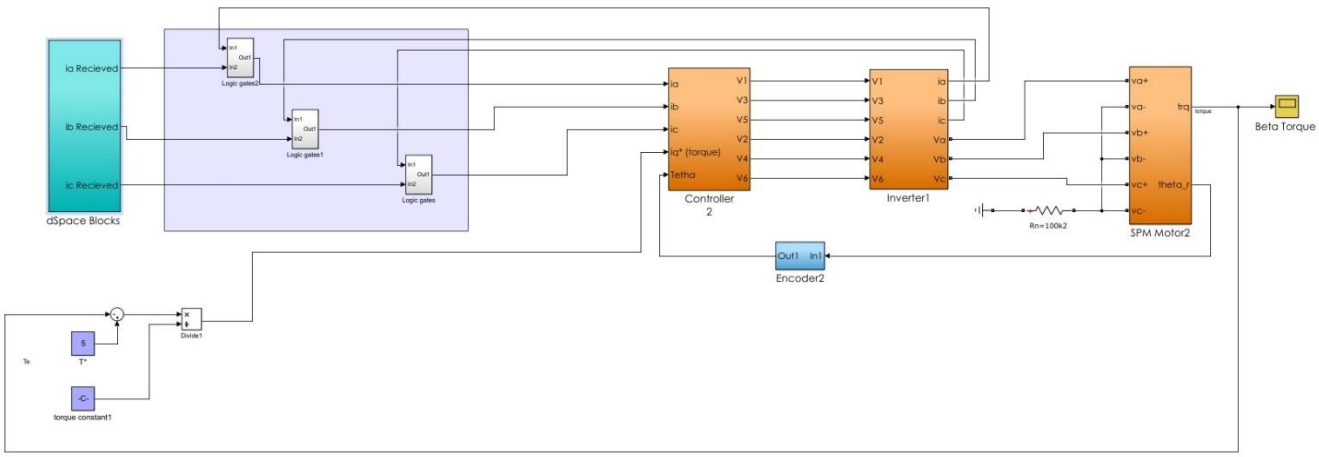

Figure 6. 5: Slave Simulink model with logic gates

As the number of samples increases in a distributed simulation, the likelihood of packet loss increases [55]. Having a system immune to packet loss allows for higher sampling rates and the possibility of transferring more dynamic signals. Figure 6.6 represents the calculated torque by the slave and master models and shows the effectiveness of the method to deal with packet loss within the communication network.

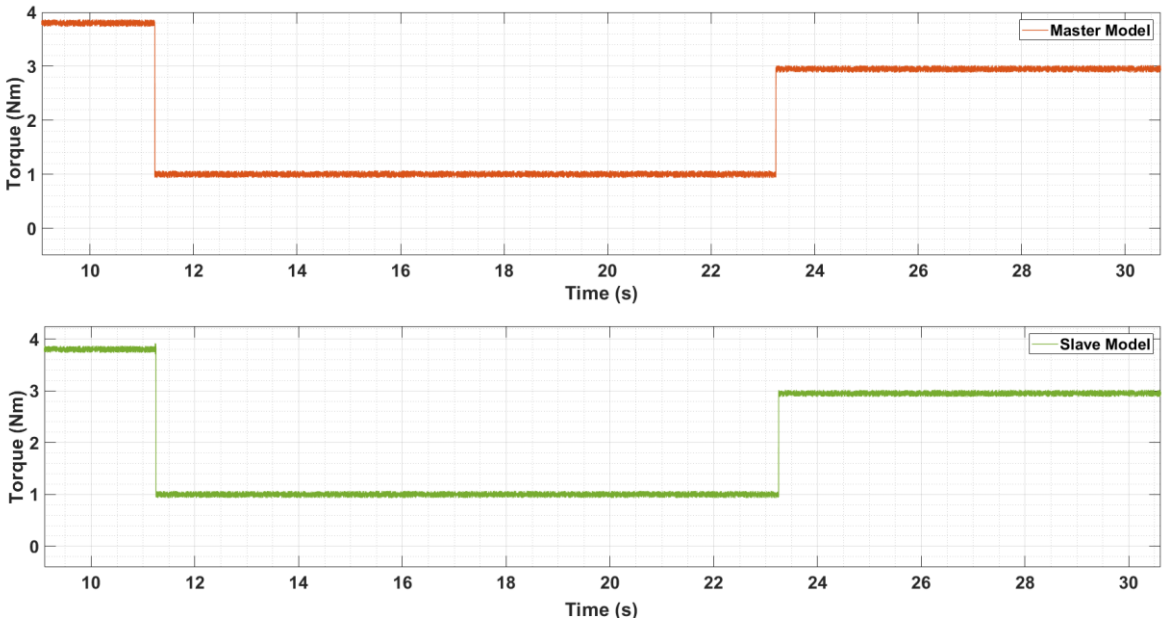

Figure 6. 6: Slave torque (green) master torque (orange) at 20% packet loss, flow chart implemented (Controldesk)

Figure 6.7 presents processed data of the slave model current signal from ControlDesk during a 20% packet loss simulation. Despite slight signal distortion in the form of small spikes caused by the blending of local and received data during a missing packet, the signal remains in a satisfactory profile, as validated by the resulting torque in Figure

6.6. It is important to note that extensive filtering was necessary to provide the findings because of the large number of samples.

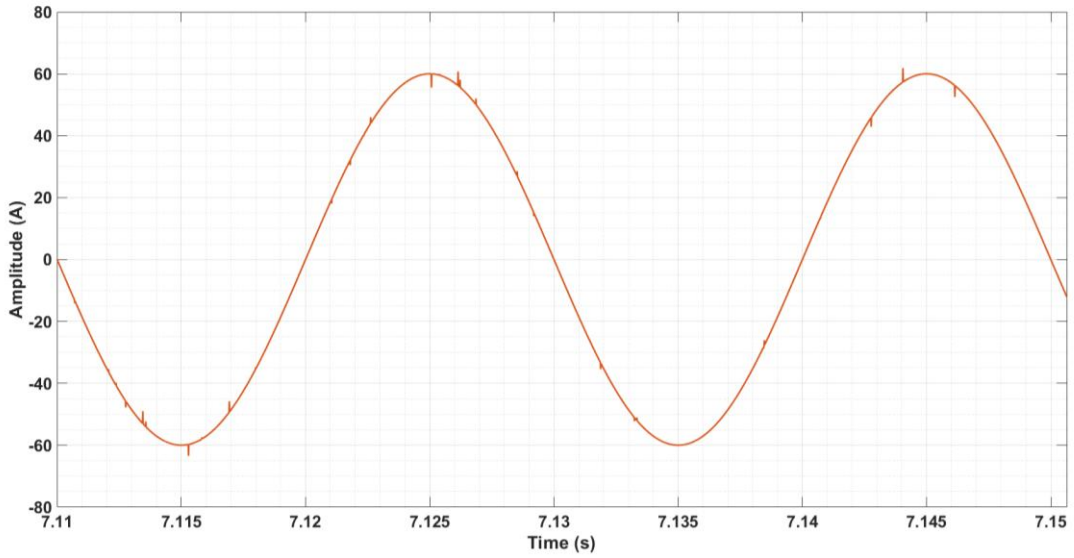

Figure 6. 7: Phase A of the three-phase current signal (20% Packet loss - ControlDesk)

In this section, we attempt to emphasize hardware-in-the-loop simulation by using the test rig's current feedback signal to run the Simulink model of the rig as a slave. Similar to section 5.2.1, we process the current signal from the hardware in ControlDesk, applying packet loss to it before we process the signal through the architecture in Figure 6.4 and then into the slave model.

Since the load machine (dyno) was unpowered, three 4Ω power resistors were connected to the star windings of the load machine to apply a resistive load to the machine under test. The industrial measurement torque sensor on the test rig measures 3.01 Nm of torque, and Figure 6.8 shows the oscilloscope snapshot. Appendix 10 represents the torque sensor.

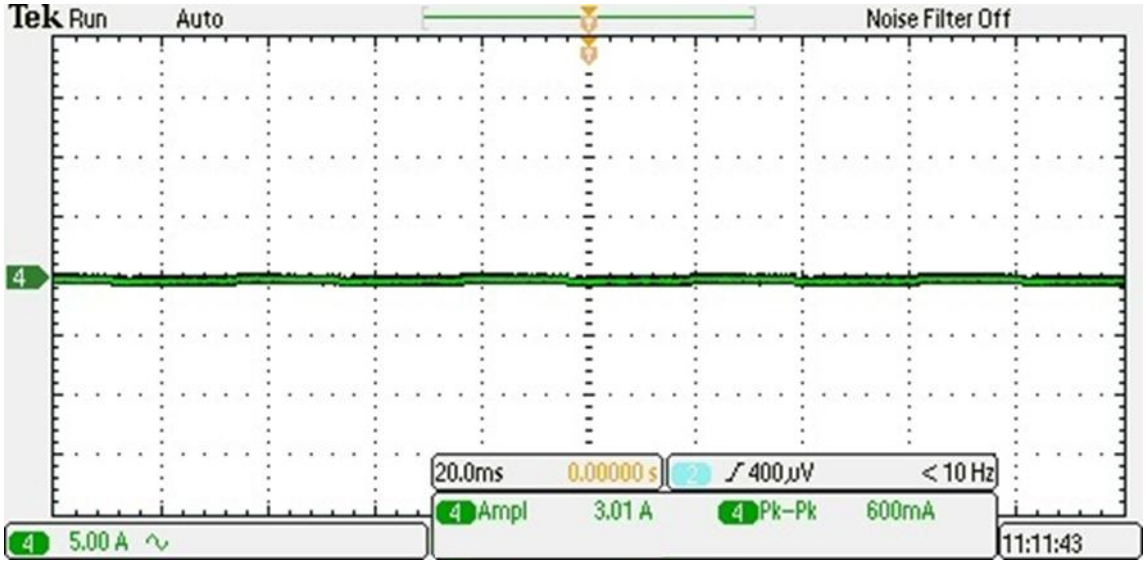

Figure 6. 8: Test rig torque measurement signal

We assume that the machine under test operates at approximately 65% efficiency at 1020 rpm. Considering [13], as well as the input voltage of 12V and 39.6A DC current from the power supply, equation (35) calculates the output power of the machine under test to be 308.9W. Equation (36) calculates the torque from the output power to be 2.9 Nm, which closely matches the measured torque from the torque sensor. [33]

$$Efficiency = \frac{P_{out}}{P_{in}} \quad (35)$$

$$P = T.\omega \quad (36)$$

The received packets combined with the local values generate a relatively flowing three-phase current signal, and the model is capable of generating a more stable torque signal, depicted in Figure 6.9. The slight difference in the slave model and the test rig-generated torque value, ≈0.2 Nm, is due to the lower fidelity of the model compared to the actual setup, where there are further considerations, including mechanical and electrical losses.

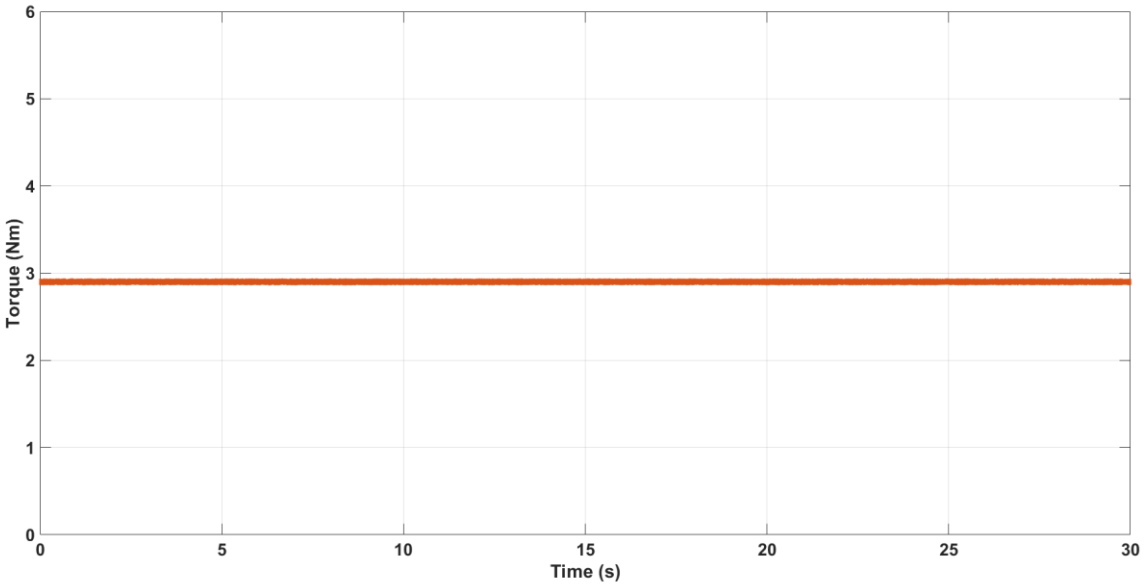

Figure 6. 9: Torque Signal After processing the packet compensation blocks.

This section's approach successfully demonstrated the simulation setup's packet loss immunity and its effective open-loop operation. In industrial environments, there are various situations where such setups can be helpful in improving cost and time efficiency. For instance, we need to test a new tune for an electric power pack (EPP) that includes the electronic control unit (ECU) and the motor, or we need to compare the performance of two different motors that share the same ECU unit. In addition, this section is a fundamental step toward having a reliable, full range, distributed electric drive system simulation, which is addressed in the next section.

6.2 Close-Loop Setup: EM Model in The Loop

This section focuses on achieving the main objective of the thesis, which involves a close-loop Sil simulation, as illustrated in Figure 6.2. As previously explained, this setup operates the electric motor (slave model) using power inputs from the master model, which then feeds the achieved torque signal back into the master.

Furthermore, instability in the system, manifesting as jitter, packet loss, latency, connection error, or simulation discontinuation, renders the controller incapable of maintaining functionality. In the case of hardware in the simulation, this instability can lead to damage to the hardware and exacerbate safety issues.

The approach discussed in the previous section proved to be effective for packet loss in open-loop simulation. However, it requires a more robust approach to address other elements, such as latency and jitter.

After conducting a series of simulations in a closed loop, we discovered that as the system's latency increases up to approximately 20 ms, the locally generated torque from the master model becomes increasingly distinct from the torque feedback, leading to complete system instability, especially if a change is introduced in the load; we implement this relationship in Table 6.1.

Distance	Network Latency	Torque (error) (subtraction of master and slave resulting torques)
0 km	0 ms	0 Nm
	10 ms	0.09 Nm
	20 ms	0.18 Nm
	30 ms	0.27 Nm
	40 ms	0.36 Nm
	50 ms	0.45 Nm
	60 ms	0.54 Nm
	70 ms	0.63 Nm
	80 ms	0.72 Nm
	90 ms	0.81 Nm
	100 ms	0.9 Nm
	110 ms	0.99 Nm
	120 ms	1.08 Nm
	130 ms	1.17 Nm
	140 ms	1.26 Nm
	150 ms	1.35 Nm
	160 ms	1.44 Nm
	170 ms	1.53 Nm
	180 ms	1.62 Nm
	190 ms	1.71 Nm
	200 ms	1.8 Nm
	210 ms	1.89 Nm
	220 ms	1.98 Nm
	230 ms	2.07 Nm
	240 ms	2.16 Nm
	250 ms	2.25 Nm
	260 ms	2.34 Nm
	270 ms	2.43 Nm
	280 ms	2.52 Nm
	290 ms	2.61 Nm
10000 km	300 ms	2.7 Nm

Table 6. 1: Torque feedback error (Nm) vs. latency (ms)

the association between the two resultant torque and latency presented in Table 6.1 has been linearised in the plot of torque error versus latency in Figure 6.10; this linearisation allows for estimation of the latency within the setup. By means of plotting the feedback error against latency, slope equation can be developed so that latency can be calculated with the error in the torque feedback signal. Figure 6.10 depicts error in the feedback signal against latency.

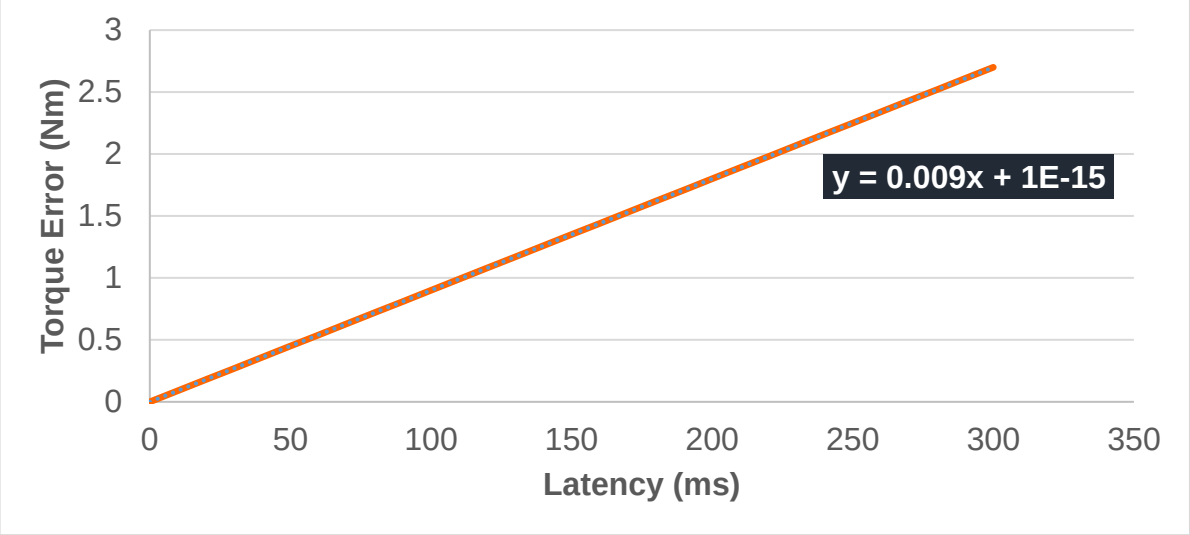

Figure 6. 10: Torque feedback error (Nm) Vs. latency (s) curve

Constructed from the slope equation displayed on Figure 6.10, latency in the system can be estimated under different test conditions, and action can be taken accordingly to maintain real-time operation and avoid any discontinuation in the system under different assessments. The simulation becomes impermanent when there are changes in the load, there is a disconnection between the nodes, or the network is experiencing latency.

Both the master and slave models undergo modifications in response to the proposed latency estimation method and to implement the developed communication architecture, depicted in Figure 6.11, that allows for stable and robust real-time distributed Sil simulation.

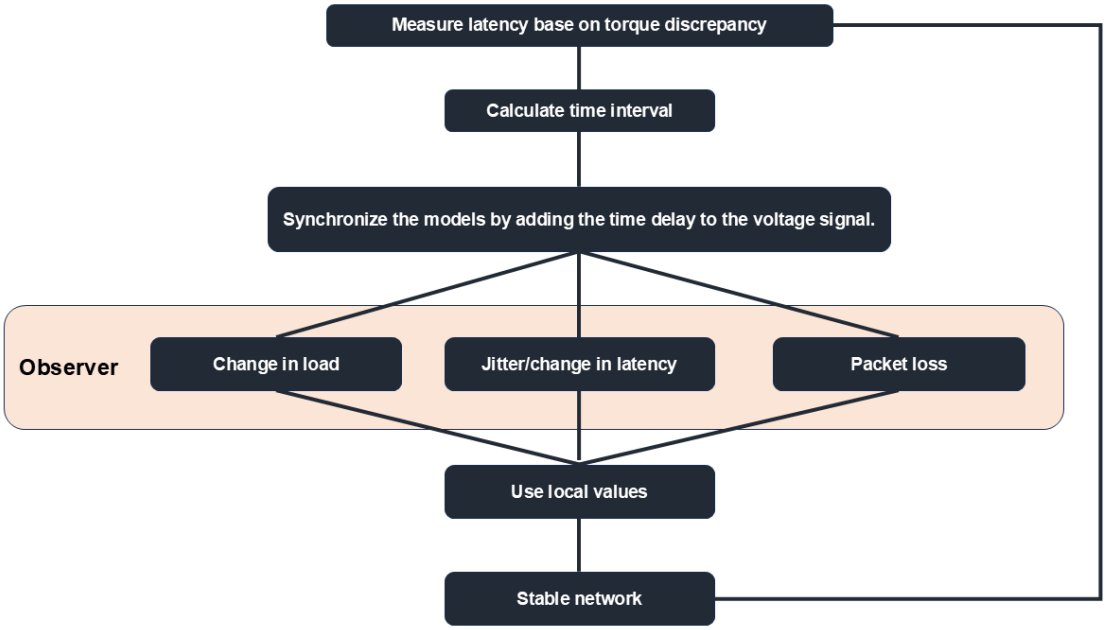

Figure 6. 11: Close-loop Sil communication architecture

Based on this architecture, the master model now comprises two new sets of Simulink blocksets, with one serving as an observer and the rest acting as a compensator for network impairment. Upon running the simulation, the latency is calculated based on the torque difference between master and slave. Next, we compute the time interval, which synchronises the models by coupling them on the same cycle. Equation (37) calculates the time delay for the voltage signal, considering the estimated latency.

$$N = \frac{120 \times f}{P} = \frac{120 \times \frac{1}{T}}{P} \qquad (37)$$

This scenario entails a 20 ms network latency while the drive system functions at 1000rpm.The resultant time interval is 15 ms. Factoring in a turnaround and a model calculation time of 0.4 ms results in a total of 40.4 ms. Taking into account the time interval, this overall time corresponds to a modulus of 0.69. Converting this within the time interval yields 10.4 ms, necessitating the introduction of a 19.6 ms time delay to the voltage signal, as indicated in equations (38) and (39).

$$40.4 - 10.4 = 30ms \qquad (38)$$

$$Timer\ set\ to = (30 \times 2) - 40.4 = 19.6ms \qquad (39)$$

The observer detects simulation discontinuity due to jitter, packet loss, or sudden changes in the load. When jitter and packet loss occur, the observer uncouples the models and uses local inputs for both. However, only the torque feedback disconnects

when the load changes, enabling the slave model to catch up with the master model. As discussed in the previous chapter, in case of inaccurate feedback, there is a risk of instability in the system and, in case of hardware, serious defection risk. Once the network stabilises, we estimate the latency and apply a time delay based on latency to synchronise the models before coupling the models again. Figure 6.12 represents the overall master model blocks, and Figure 6.13 and Figure 6.14 represent architecture implementation at different stages, respectively.

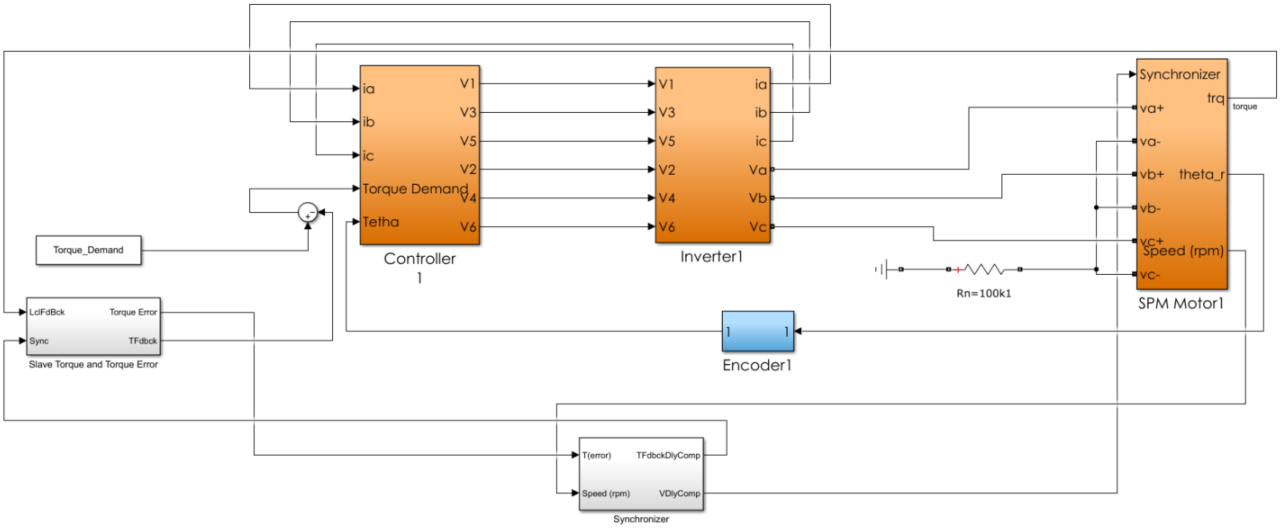

Figure 6. 12: Master model overview

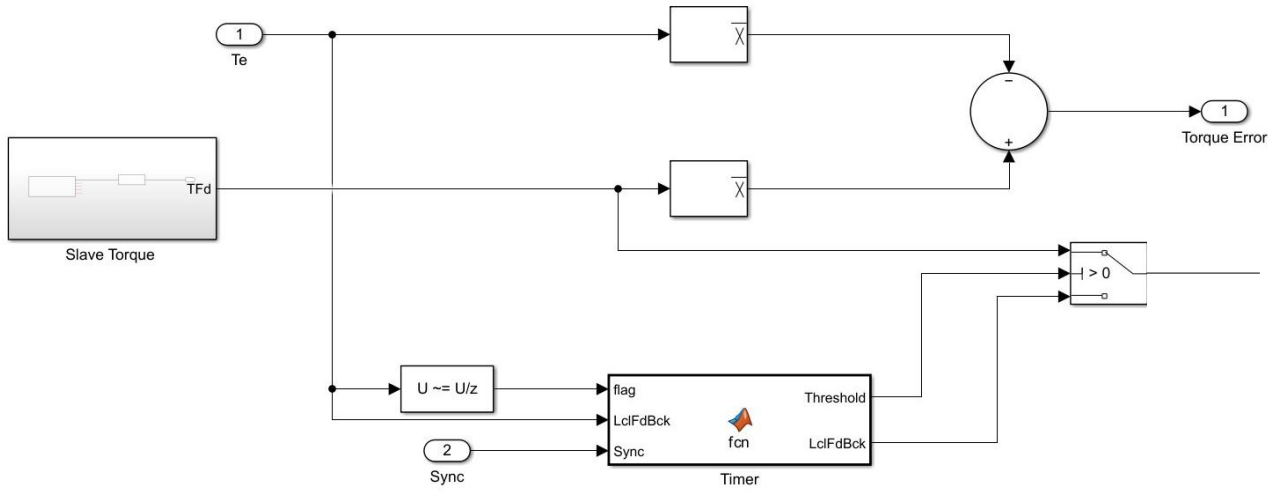

Figure 6. 13: Torque feedback signal correction in master model

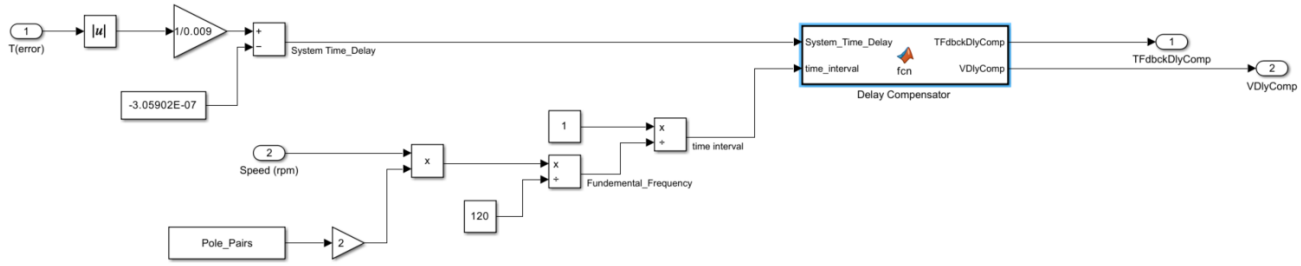

Figure 6. 14: Simulink block to determine master model timer

Moreover, the blocks included into the slave model adhere to the same notion as outlined in the preceding section. A local duplicate of the model was established in the slave model, which operates on the torque demand value from the master model and retains the same value if a new packet is not received. The average voltage signal from the master model is compared to the previously sent average voltage to identify any significant variations that may indicate communication or packet loss. Furthermore, the model alternates between the incoming packets and local packets to ensure functionality. This method enables constant voltage input to the motor, hence preventing simulation errors or potential damage to the motor in the presence of hardware inside the simulation. To enhance this process, a lookup table or reduced-order model can be employed to substantially increase computing efficiency for a higher fidelity model, Figure 6.15 illustrates the additional blocks included into the slave model to execute the methodology.

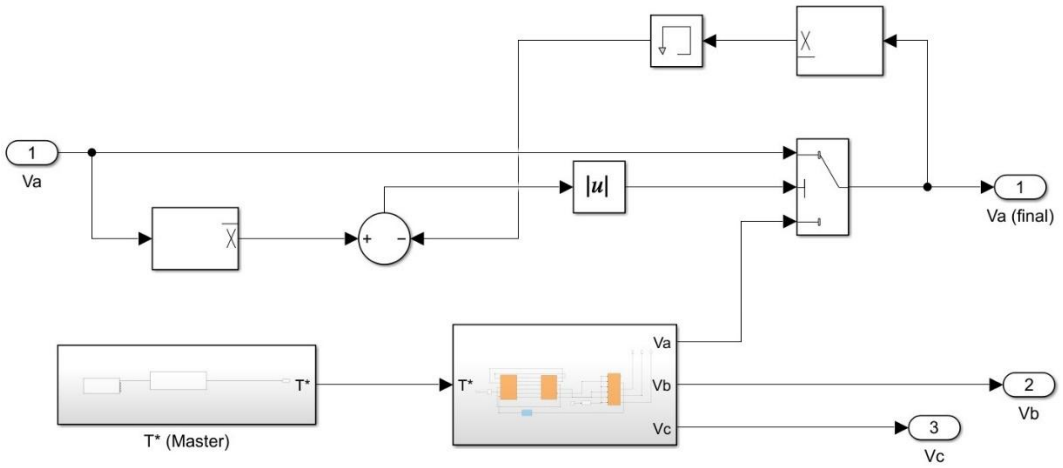

Figure 6. 15: Processing of received voltage signals in slave model

The simulation configuration is assessed on Scalexio with two test cases to validate the methodology. During the first test instance, the network demonstrated 20 ms of

jitter and 10% random packet loss. Figure 6.16 illustrates that the system can maintain a steady condition throughout the trial.

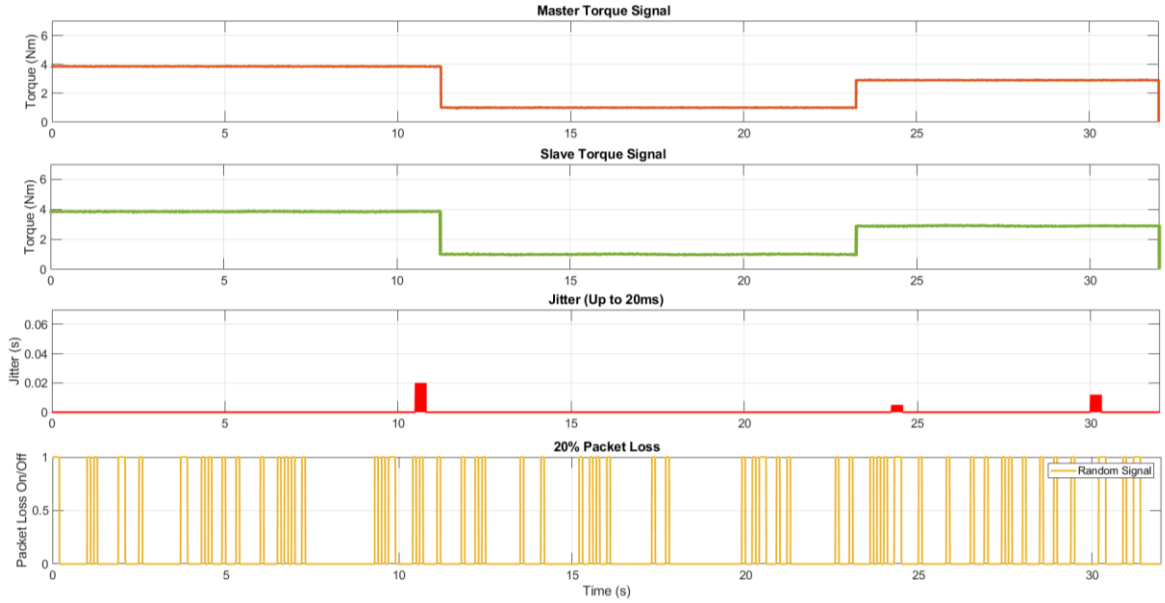

Figure 6. 16: 20 ms jitter, 10% packet loss

In the second scenario, we increase the jitter to 50 ms while maintaining a packet loss of 10%. Figure 6.17 demonstrates that the system has considerable robustness under these testing circumstances. However, an oscillation in the torque signal especially after each latency change, that suggests that the latency estimation approach needs improvement for use in significantly unstable test scenarios. The network overview in chapter five outlines the rare occurrence of this test scenario. Notwithstanding this, the system remains quite stable.

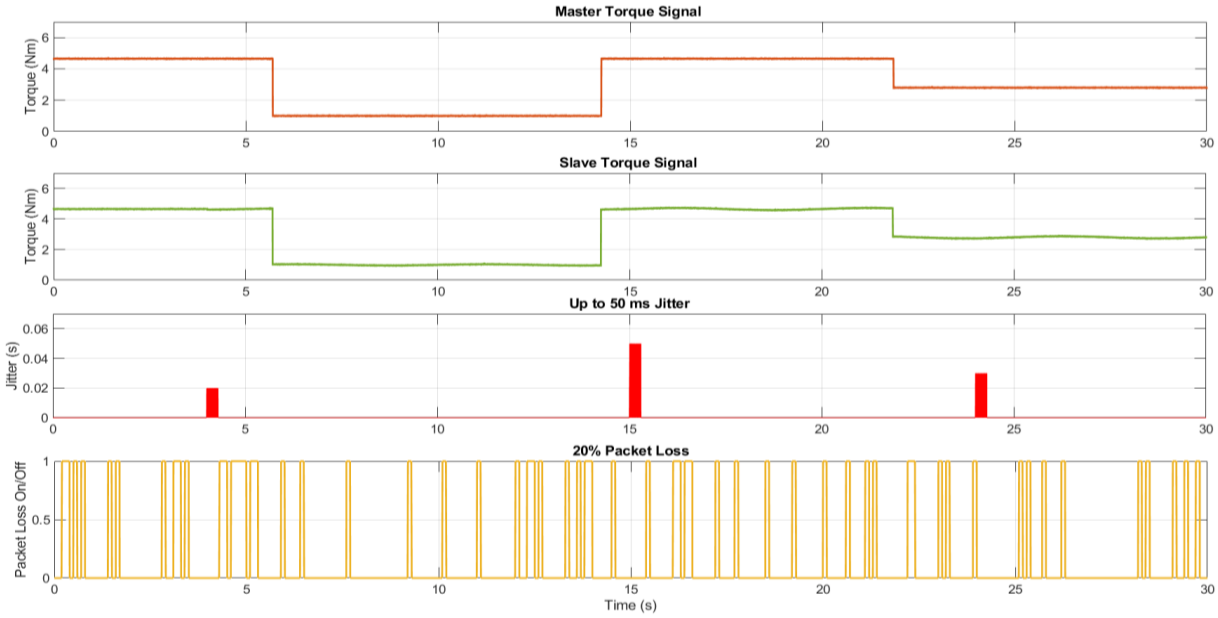

Figure 6. 17: 50ms jitter, 20% packet loss test case.

This section introduced a novel communication architecture based on a developed approach for estimating network delay. The approach was verified by two distinct test cases in a closed-loop software-in-the-loop simulation. The results indicate that the configuration is very resilient and can sustain stability under various network degradation situations.

6.3 Summary

This chapter effectively illustrated a novel and customized method for activating a distributed version of X within the loop simulation. The approach addresses packet loss by enabling local input into the model when the observer detects missing packets or sudden changes. Furthermore, the system estimates latency using torque feedback from both the slave and master models, enabling delay injection to synchronize the two models.

The architecture developed in this chapter, depicted in Figure 6.11, allows the setup to maintain stability during network faults and discontinuities, as well as the temporary nature of the network or simulation. The slave model, like the master, is utilizing local packets under these conditions.

This chapter presented a methodology that successfully demonstrated the possibilities of incorporating a dynamic and complex system, like an electric machine, into a distributed XiL setup. We successfully conducted the final simulation setup, as depicted in Figure 6.2, and provided an accompanying explanation for the results. This research has effectively provided fundamental and pivotal knowledge to take distributed real-time simulation to even more challenging and dynamic systems with a higher level of fidelity.

Chapter 7

Conclusions and further work

7.1 Summary of findings

The automotive industry is moving toward electrification, which has increased the complexity of powertrain systems within a vehicle. Developing a new vehicle requires precise design and investment. Considering the high number of components in a powertrain system, the process of integrating major subsystems to form a successful product is complex, challenging, costly, and time-consuming. [56]

Traditionally, subsystem components were designed and manufactured individually and then integrated in the final stages of a vehicle's development. Despite precision pre-testing, sub-systems required minor or major modification due to sub-optimal performance in the integration phase. However, state-of-the-art development procedures now rely upon hardware-in-the-loop (HiL) techniques to improve powertrain system development time, cost-effectiveness, and safety. HiL employs a real-time simulator with a software-based model to test hardware subsystems. This removes the need for a complete physical system to be present during testing. [56]

HiL has been in use for decades, particularly in complex systems like automotive applications. Many key features make HiL an attractive technique. Replacing hardware elements of a system with software-based models significantly reduces the cost of testing and integration and allows for rapid prototyping, consistent test repeatability, and scenario/condition-based testing [18]. Furthermore, HiL can be a non-destructive approach, which allows robust testing to take place in a safe environment.

HiL has also proven to be reliable and capable of handling high-fidelity simulation. To achieve a successful high-fidelity HiL simulation, there are certain requirements to consider. According to [18], there are eight key requirements to consider.

- A computationally rapid and capable processor enables a real-time operating system.
- Signal conditioning and digital signal processing.
- Actuator fidelity and high sensor bandwidth.
- Optimized modelling. Fixed-step integration is a requirement for effective HiL.
- Virtual model stiffness.
- There is a synergy between hardware prototypes and HiL models.

- Particularly in highly dynamic systems, swift and dependable communication networks are crucial.
- Advanced laboratory instrumentation and diagnostic systems.

The distributed version of HiL, also known as Internet HiL simulation, is a new and trending topic in automotive. The significant improvement in the Internet of Things (IoT) has led to an ambition to place setups in various locations. Having a robust simulation setup for different subsystems can significantly improve cost and time efficiency by starting the integration and prototyping process from an earlier stage of development.

Within the automotive industry, original equipment manufacturers, or OEMs, are investing heavily in digital twins and distributed software and hardware simulation. This thesis served as a feasibility study for the implementation of a dynamic system, specifically an electric machine, within a physically distributed simulation environment.

The communication aspect of distributed hardware in the loop simulation (D- HiL) is a significant challenge. Factors such as model fidelity, coupling points, communication architecture, protocol, and observers and predictors significantly influence the results. TCP is the best choice for slower systems, while UDP is suitable for dynamic systems due to its flexibility and low protocol overhead. Communication architecture also plays a significant role in results. The electric drive system poses significant challenges, including data transfer and a customized communication requirement. This thesis has developed a novel approach that considers the user datagram protocol, communication architecture, error prediction, and error rectify methods to meet the dynamic requirements of electric machine drive systems. [57]

The feasibility study was done over four phases to both investigate the effect of network impairment and put the proposed network architecture to the test. Figures 7.1 to 7.5 represent the different phases of this thesis.

Figure 7. 1: Hardware and virtual model components for the acquired test rig

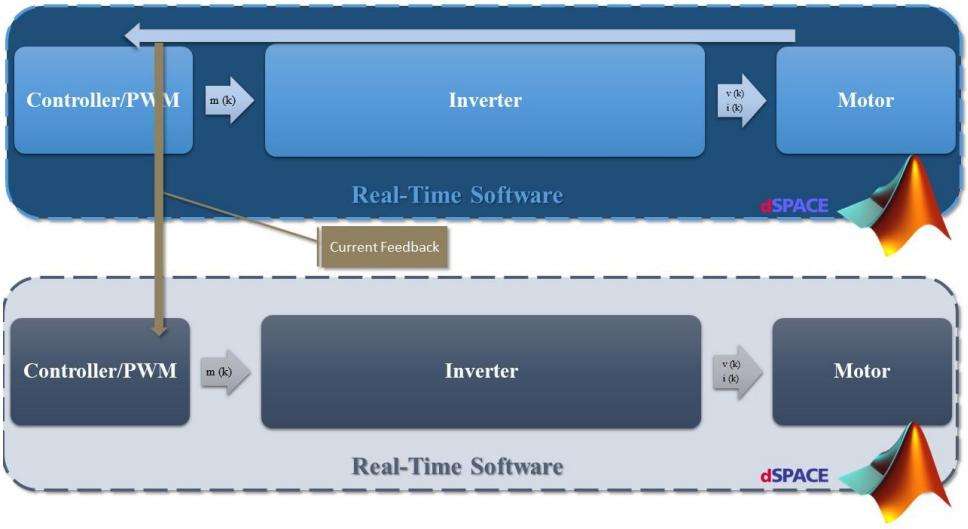

Figure 7. 2: The second phase of the project distributed real-time open-loop software-in-the-loop.

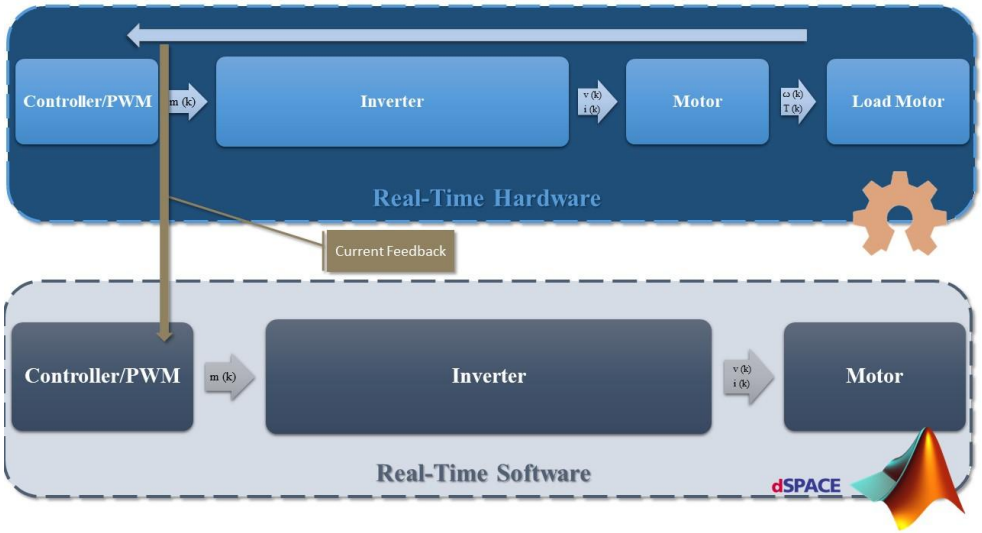

Figure 7. 3: The second phase of the project distributed real-time open-loop hardware-in-the-loop.

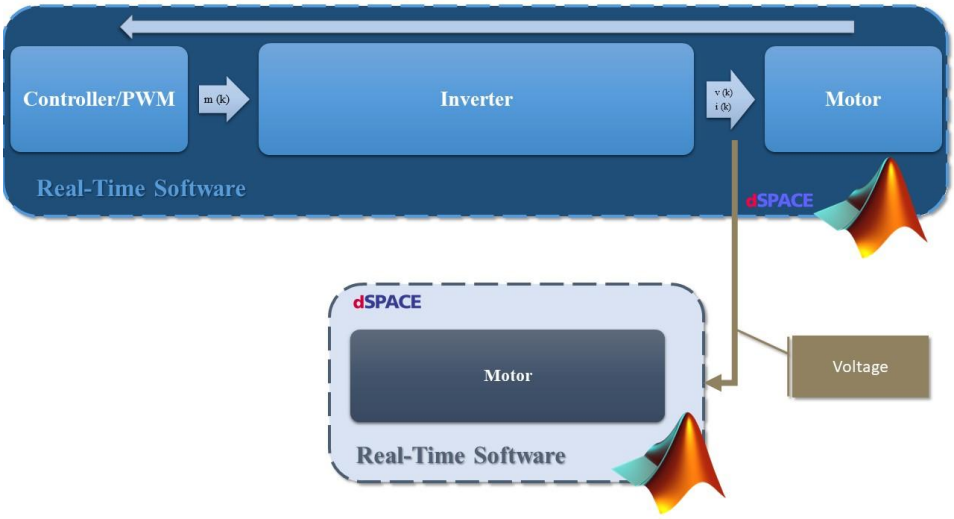

Figure 7. 4: The fourth phase of the project, distributed real-time open-loop software-in-the-loop.

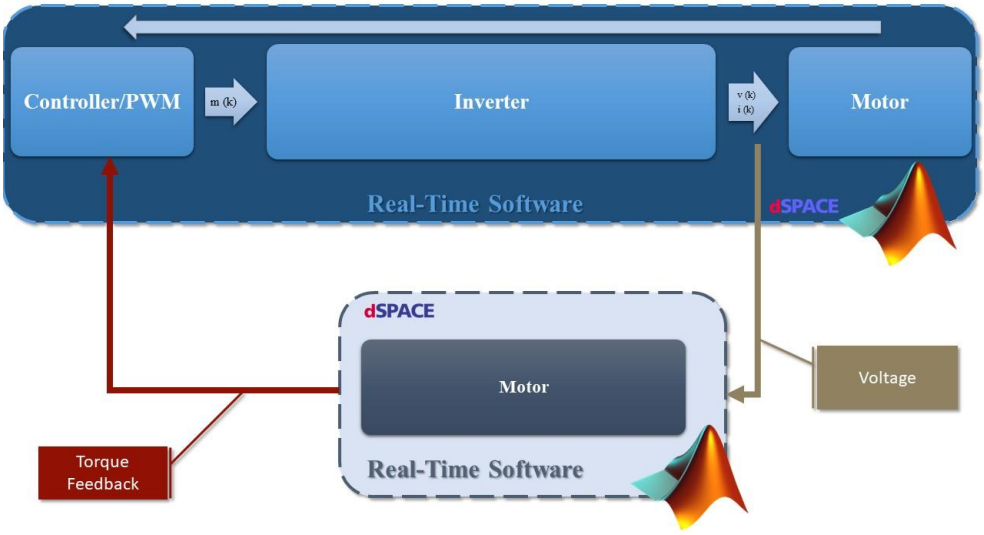

Figure 7. 5: Final Stage of The Project, Distributed Real-Time Hardware-In-The-Loop

The first objective was to establish a digital twin of the test rig employed in this thesis, which consists of an electric motor, power electronics, and a controller, depicted in Figure 7.1.

The Simulink model was used to analyse the impact of network impairments on dSpace and MATLAB Simulink, including latency, jitter, and packet loss. Changes in communication node and setup affect the system differently, but they all cause instability and significantly impact performance. Packet loss leads to control instability, uncontrollable motor torque, vibrations, and overheating, causing emergency system shut down and hardware damage. Latency results in losing synchronization between control and motor response, leading to hardware damage and overshoot. Jitter results

in severe torque pulsations, excessive stress on mechanical components, and hardware damage due to high variability in data timing.

Furthermore, a new method for activating a distributed version of X within loop simulation, addressing packet loss, jitter and latency using torque feedback from both slave and master models. The architecture maintains stability during network faults and discontinuities, using local packets for both models. The methodology successfully incorporates dynamic systems like an electric machine into a distributed XiL setup.

The next section highlights the contribution of this thesis to preexisting knowledge and the value it has added to distributed HiL in the automotive industry.

7.2 Contributions to Knowledge

Numerous studies have been conducted on distributed XiL in several fields, including smart grids, robotics, gaming, and others. An electric drive system requires a high sample rate, necessitating a robust CPU for real-time simulation, along with a significantly larger packet size of approximately 20 Mb at 16-bit precision, hence demanding greater bandwidth than other applications. This thesis conducted a feasibility assessment that classified the effects of several network issues and emphasized their potential to disrupt the system. Moreover, the observer and communication architecture delineated in this thesis have effectively fulfilled the objectives specified in the proposal and may serve as a significant resource for future research.

7.3 Implications of the Research

This thesis represents a significant advancement toward achieving a comprehensive distributed hardware-in-the-loop simulation for electric drive systems. Besides the benefits previously mentioned in this thesis, the demonstrated methodology's implications open up numerous additional ambitious benefits.

- Remote Monitoring and Control: This technology enables real-time observation, diagnostics, and management from various geographical locations.
- Real-Time Data Collection: This feature enables continuous data logging for performance evaluation, defect identification, and system enhancement.
- Predictive Maintenance: Predictive algorithms, in conjunction with the proposed communication architecture operating over the internet, can assess machine performance and prevent breakdowns.

- Research and Development: Using this methodology can enhance the testing, prototyping, and validation processes.
- Remote software update and validation.

7.4 Research Limitations and Challenges

Despite the fact that the findings of this research are fundamental and solid progress toward the implication of real-time distributed hardware in the loop simulation, the novel methodology explained in this thesis has its own limitations. One of these limitations is the need for a duplicate model in both the master and slave models, which is not always available. Therefore, future work, as we will discuss in the next section, may employ lower-fidelity duplicates of the models or lookup tables to replicate the actual system.

Moreover, sudden changes in the system's operation can be a requirement for some test scenarios, such as conducting a dynamic test; the methodology is not best in this regard, and it needs refinement.

In addition, this research project has undergone numerous changes since its inception, leading to a significant loss of time. At the start of the project, a new supervisory team replaced the initial one, leading to changes in the original project proposal. Financial difficulties led to the exploitation of funding opportunities and the establishment of an industrial contribution to tailor the project to the company's needs. However, Brexit-related financial issues ultimately led to the withdrawal of the industrial contribution. The proposal underwent further changes after six months of coordination with the company's needs. Furthermore, the COVID-19 pandemic was the major negative influence in this project, as the dSpace equipment was only accessible from the lab. In addition to being unable to visit the lab for approximately a year and a half, the disease struck me three times, necessitating a ten-day hospital stay on the first occasion. Fortunately, the wonderful NHS staff helped me recover. I endured the lingering effects of the disease for several months. The mental challenges due to financial situations and lack of progress in the project as a result of the pandemic resulted in stress and anxiety that forced me to get a two-month interruption. Unforeseen issues continued to delay the research project, leading to funding expiration and further financial difficulties. Consequently, I accepted a job offer from ZF Group in June 2022, which compelled me to relocate to Birmingham in September. Moreover, technical challenges related to setting up the dSpace simulator as well as working with the IT team at

Newcastle University were significantly time-consuming. Service requests to dSpace frequently took a significant amount of time to resolve.

7.5 Recommendations for Future Work

This thesis's feasibility study yielded robust results on the effects of network impairments and confirmed the viability of distributed XiL for electric drive systems. Additionally, this research may spark many future research opportunities, some of which are listed below.

- Employing diverse connection methods like 5G networks.
- Investigating the industrial potential of the developed testing philosophy.
- Introducing further sub-systems into the test setup.
- Apply the test concept in a more practical setup.
- Develop a reduced-order model or lookup table for the slave model compensation blocks.
- Testing the approach both with higher fidelity models and at the hardware level.

Reference

[1] C. Toyota Motor, "Sales, Production, and Export Results for the First Half of 2023," *Toyota Global*, 2023 2023. [Online]. Available: https://global.toyota/en/finance/.

[2] A. G. Volkswagen, "Financial Reports and Publications," *Volkswagen Investor Relations*, 2023 2023. [Online]. Available: https://www.volkswagenag.com/en/InvestorRelations.html.

[3] M. General, "Earnings Releases and Reports," *GM Investor Relations*, 2023 2023. [Online]. Available: https://investor.gm.com/.

[4] C. Ford Motor, "Financial Statements," *Ford Investor Relations*, 2023 2023. [Online]. Available: https://corporate.ford.com/investors.html.

[5] L. Honda Motor Co, "Financial Results and Reports," *Honda Global*, 2023 2023. [Online]. Available: https://global.honda/investors/library.html.

[6] F. Brown, "Biggest Car Manufacturers by Revenue, 2023," *CEOWORLD magazine*, 2023/10/25 2023. [Online]. Available: https://ceoworld.biz/2023/10/25/report-biggest-car-manufacturers-by-revenue-2023/.

[7] statista. "Global automotive market size 2030 | Statista." https://www.statista.com/statistics/574151/global-automotive-industry-revenue/ (accessed.

[8] S. Sen, "Automotive powertrain development: virtually-connected hardware co-simulation," *SAE International: MOBILITY ENGINEERING*, vol. 4, no. 2, pp. 38-39, 2017.

[9] S. Almdawar, "Developing a Fast Computing Second-Order Transfer Function Engine Model Using System Identification for Internet Distributed Hardware in the Loop Application," 2021/12/13 2021.

[10] C. National and A. Telecommunications, "Latency Measurement: What is Latency and How Do We Measure It?," 2020. [Online]. Available: https://www.ncta.com/what-we-do/latency-measurement

[11] BroadbandSpeedTest.org.uk. "Broadband Speed Test." https://www.broadbandspeedtest.org.uk/ (accessed.

[12] Jisc. "Janet Network." https://www.jisc.ac.uk/janet (accessed.

[13] A. A. Hassan, "A fault tolerant motor drive for electric power steering systems," Newcastle University, 2019. [Online]. Available: http://theses.ncl.ac.uk/jspui/handle/10443/4792

[14] J. Blyler and Society of Automotive Engineers, *Software-hardware integration in automotive product development*, 2014, pp. 1 PDF (v, 113 pages). [Online]. Available: http://digitallibrary.sae.org/content/pt-161.

[15] T. B. M. S. A. S. J. L. F. Z. F. H. K. Ersal, "Development of an Internet-Distributed Hardware-in-the-Loop Simulation Platform for an Automotive Application," 2009/10/12 2009: ASME, pp. 73-80. [Online]. Available: https://doi.org/10.1115/DSCC2009-2709. [Online]. Available: https://doi.org/10.1115/DSCC2009-2709

[16] Z. S. F. H. K. Fathy, J. Hagena, and J. L. Stein, "Review of hardware-in-the-loop simulation and its prospects in the automotive area," *Modeling and simulation for military applications*, vol. 6228, International Society for Optics and Photonics, p. 62280E, 2006.

[17] X. U. Q *et al.*, "The Design of Intelligent Experimental System for Hardware-in-the-loop Simulation of Aerocraft Control System," in *2019 Chinese Control And Decision Conference (CCDC)*, 3-5 June 2019 2019, pp. 2870-2875, doi: 10.1109/CCDC.2019.8832939.

References

[18] M. Aljehani and M. Inoue, "Performance Evaluation of Multi-UAV System in Post-Disaster Application: Validated by HITL Simulator," *IEEE Access*, vol. 7, pp. 64386-64400, 2019, doi: 10.1109/ACCESS.2019.2917070.

[19] W. S. Harrison, D. M. Tilbury, and C. Yuan, "From Hardware-in-the-Loop to Hybrid Process Simulation: An Ontology for the Implementation Phase of a Manufacturing System," *IEEE Transactions on Automation Science and Engineering*, vol. 9, no. 1, pp. 96-109, 2012, doi: 10.1109/TASE.2011.2162584.

[20] J. Lima, P. Costa, T. Brito, and L. Piardi, "Hardware-in-the-loop simulation approach for the Robot at Factory Lite competition proposal," in *2019 IEEE International Conference on Autonomous Robot Systems and Competitions (ICARSC)*, 24-26 April 2019 2019, pp. 1-6, doi: 10.1109/ICARSC.2019.8733649.

[21] H. Y. Irwanto, "Development of Instrumentation, Control and Navigation (ICON) for Anti Tank Guided Missile (ATGM)," in *2016 2nd International Conference on Science in Information Technology (ICSITech)*, 26-27 Oct. 2016 2016, pp. 137-141, doi: 10.1109/ICSITech.2016.7852622.

[22] S. Noureen, N. Shamim, V. Roy, and S. Bayne, "Real-Time Digital Simulators: A Comprehensive Study on System Overview, Application, and Importance," *International Journal of Research and Engineering*, vol. 4, pp. 266-277, 12/13 2017, doi: 10.21276/ijre.2017.4.11.3.

[23] H. K. F. Z. S. H. J. S. J. L. Fathy, "Review of Hardware-in-the-Loop Simulation and Its Prospects in the Automotive Area," 2006/05/22, p. 62280E, doi: 10.1117/12.667794. [Online]. Available: https://doi.org/10.1117/12.667794

[24] Y. Zhang, S. Lu, Y. Yang, and Q. Guo, "Internet-Distributed Vehicle-in-the-Loop Simulation for HEVs," *IEEE Transactions on Vehicular Technology*, vol. 67, no. 5, pp. 3729-3739, 2018, doi: 10.1109/TVT.2018.2803088.

[25] V. Schreiber *et al.*, "Shared and Distributed X-in-the-Loop Tests for Automotive Systems: Feasibility Study," *IEEE Access*, vol. 6, pp. 4017-4026, 2018, doi: 10.1109/ACCESS.2017.2789020.

[26] T. G. R. B. Ersal, "Effect of Coupling Point Selection on Distortion in Internet-Distributed Hardware-in-the-Loop Simulation," *IEEE Transactions on Industrial Electronics*, vol. 60, no. 4, pp. 1438-1445, 2013, doi: 10.1109/TIE.2012.2205859.

[27] V. Schreiber, K. Augsburg, V. Ivanov, and H. Fujimoto, "Novel Developing Environment for Automated and Electrified Vehicles using Remote and Distributed X-in-the-Loop Technique," in *2020 IEEE Vehicle Power and Propulsion Conference (VPPC)*, 18 Nov.-16 Dec. 2020 2020, pp. 1-5, doi: 10.1109/VPPC49601.2020.9330996.

[28] H. Tandoh, Y. Cao, and A. Awoyomi, "Smith predictor for slug control with large valve stroke time," 2017, pp. 1531-1536.

[29] T. B. M. S. J. L. F. H. K. Ersal, "Statistical Transparency Analysis in Internet-Distributed Hardware-in-the-Loop Simulation," *IEEE Transactions on Mechatronics*, vol. 17, no. 3, pp. 513-521, 2012, doi: 10.1109/TMECH.2012.2194841.

[30] K. Clark. "How Many Electric Motors Are in Your Car?" https://www.example.com/how-many-electric-motors-are-in-your-car (accessed.

[31] M. P. P. Harrer, *Steering Handbook*. SpringerLink, pp. XVII-565.

[32] S. Magnus. "The Key Differences Between An Electric vs. Hydraulic Power Steering." https://www.example.com/electric-vs-hydraulic-power-steering (accessed.

[33] J. R. Hendershot and T. J. E. Miller, *Design of brushless permanent-magnet machines*, 2nd edition ed. Venice, Florida: Motor Design Books, 2010, pp. xxiv, 798 p.

[34] A. Vagati, G. Pellegrino, and P. Guglielmi, "Comparison between SPM and IPM motor drives for EV application," in *The XIX International Conference on Electrical Machines - ICEM 2010*, 6-8 Sept. 2010 2010, pp. 1-6, doi: 10.1109/ICELMACH.2010.5607911.

[35] K. H. Nam, *AC Motor Control and Electrical Vehicle Applications*. CRC Press, 2018.

References

[36] dSpace. "dSPACE: Real-time Simulation and Hardware-in-the-Loop Testing." https://www.dspace.com/en/inc/home/company.cfm (accessed.

[37] D. He, E. Hou, G. Liu, and J. Yu, "Design on Test Platform of BMS System Based on dSPACE," in *2017 International Conference on Computer Technology, Electronics and Communication (ICCTEC)*, 19-21 Dec. 2017 2017, pp. 968-971, doi: 10.1109/ICCTEC.2017.00214.

[38] A. S. K. Bandi and P. Vooka, "A new measurement technique to determine the Amplitude and Phase shift of a sinusoidal voltage signal using a charge balanced digitizer," in *2022 IEEE International Instrumentation and Measurement Technology Conference (I2MTC)*, 16-19 May 2022 2022, pp. 1-6, doi: 10.1109/I2MTC48687.2022.9806619.

[39] MathWorks. "Choose an ODE Solver." https://uk.mathworks.com/help/matlab/math/choose-an-ode-solver.html (accessed.

[40] S.-H. Kim, "Chapter 3 - Alternating current motors: Synchronous motor and induction motor," in *Electric Motor Control*, S.-H. Kim Ed.: Elsevier, 2017, pp. 95-152.

[41] R. Krishnan, *Permanent Magnet Synchronous and Brushless DC Motor Drives*. CRC Press.

[42] S. K. Sahoo, A. Ramulu, S. Batta, and S. Duggal, "Performance analysis and simulation of three phase voltage source inverter using basic PWM techniques," in *IET Chennai 3rd International on Sustainable Energy and Intelligent Systems (SEISCON 2012)*, 27-29 Dec. 2012 2012, pp. 1-7, doi: 10.1049/cp.2012.2223.

[43] M. L. Woldesemayat, H. Lee, S. Won, and K. Nam, "Modeling and Verification of a Six-Phase Interior Permanent Magnet Synchronous Motor," *IEEE Transactions on Power Electronics*, vol. 33, no. 10, pp. 8661-8671, 2018, doi: 10.1109/TPEL.2017.2782804.

[44] M. Elgbaily, F. Anayi, and M. M. Alshbib, "A Combined Control Scheme of Direct Torque Control and Field-Oriented Control Algorithms for Three-Phase Induction Motor: Experimental Validation," *Mathematics*, vol. 10, no. 20, doi: 10.3390/math10203842.

[45] B. T. A. G. X. T. P. R. S. L. Zigmund, "Experimental Evaluation of PI Tuning Techniques for Field Oriented Control of Permanent Magnet Synchronous Motors," *IEEE Transactions on Industrial Electronics*, vol. 61, no. 4, pp. 897-904, 2024, doi: 10.1109/TIE.2024.2043597.

[46] N. C. C. Ripe. "RIPE NCC: Réseaux IP Européens Network Coordination Centre." https://www.ripe.net/ (accessed.

[47] S. P. perf. "What is perfSONAR GTK?" https://www.perfsonar.net/gtk_whatis.html (accessed.

[48] L. Fiveable. "Throughput in Computer Networks: Study Guide." https://library.fiveable.me/computer-networks-a-systems-approach/unit-4/throughput-computer-networks/study-guide/HX91wdeQuwIlGhuV (accessed.

[49] S. P. perf. "perfSONAR Use Cases." https://www.perfsonar.net/use_cases.html (accessed.

[50] S. Guo, Y. Liu, Y. Zheng, and T. Ersal, "A Delay Compensation Framework for Connected Testbeds," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 52, no. 7, pp. 4163-4176, 2022, doi: 10.1109/TSMC.2021.3091974.

[51] O. Boubaker, V. Balas, A. Benzaouia, C. Mohamed, M. Mahmoud, and Q. Zhu, *Time-Delay Systems-Modeling, Analysis, Estimation, Control, and Synchronization*. 2017.

[52] MathWorks. "Model Wireless Message Communication with Packet Loss and Channel Failure." https://www.mathworks.com/help/simulink/model-wireless-message-communication-with-packet-loss-and-channel-failure.html (accessed.

References

[53] A. Giannakou, D. Dwivedi, and S. Peisert, "A machine learning approach for packet loss prediction in science flows," *Future Generation Computer Systems*, vol. 102, pp. 190-197, 2020/01/01/ 2020, doi: https://doi.org/10.1016/j.future.2019.07.053.

[54] E. G. W. M. D. Q. D. E. F. M. Peters, "Predictive control for networked systems affected by correlated packet loss," *International Journal of Robust and Nonlinear Control*, vol. 29, no. 13, pp. 4043-4061, 2019, doi: 10.1002/rnc.4722.

[55] L. Feng-Li, J. R. Moyne, and D. M. Tilbury, "Performance evaluation of control networks: Ethernet, ControlNet, and DeviceNet," *IEEE Control Systems Magazine*, vol. 21, no. 1, pp. 66-83, 2001, doi: 10.1109/37.898793.

[56] J. Guo, Z. Li, H. Dong, R. Li, and N. Sun, "Design of intelligent transportation verification platform based on hardware in the loop simulation technology," in *2022 8th International Conference on Hydraulic and Civil Engineering: Deep Space Intelligent Development and Utilization Forum (ICHCE)*, 25-27 Nov. 2022 2022, pp. 401-405, doi: 10.1109/ICHCE57331.2022.10042519.

[57] D. Madhuri and P. C. Reddy, "Performance comparison of TCP, UDP and SCTP in a wired network," in *2016 International Conference on Communication and Electronics Systems (ICCES)*, 21-22 Oct. 2016 2016, pp. 1-6, doi: 10.1109/CESYS.2016.7889934.

Appendix

Appendix.1 Magnetic Flux in SPM and IPM Machines

Consider the SPMSM shown in Figure 8.1. The two pictures illustrate the same motor with two distinct flux routes corresponding to the d and q phase windings. PMs are represented by dark arcs. The lines shown in Figure 8.1 (a) represent d-axis flux corresponding to d-axis current. Observe that the d-axis windings are aligned with the q-axis.

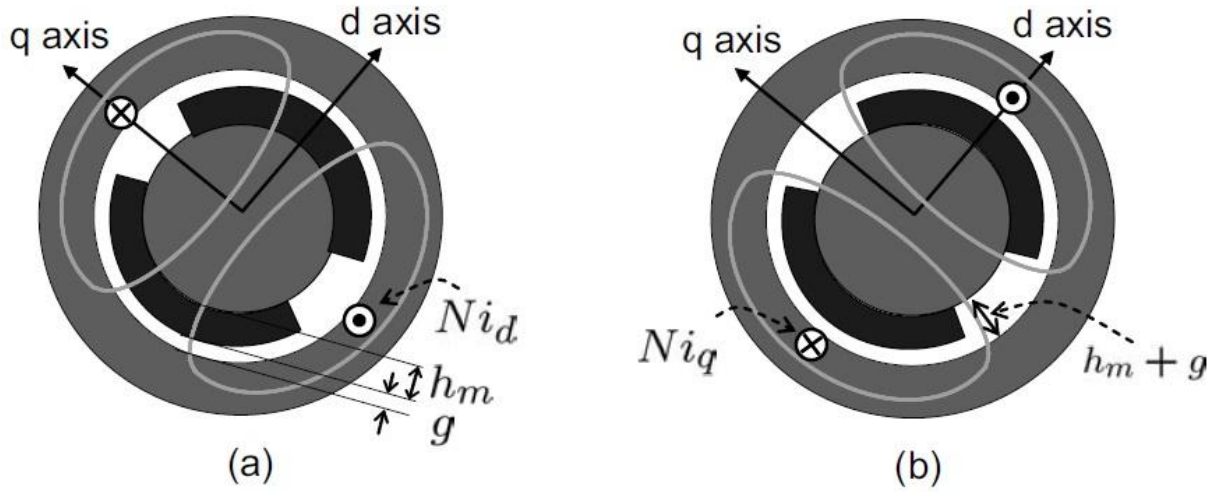

Figure 8. 1: (a) d–axis flux path, (b) q–axis flux path [33]

As a result of applying Ampere's law to the d–axis current and integrating through the indicated loop in Figure 8.1(a), it can be concluded that

$$\frac{B}{\mu PM} 2h_m + \frac{B}{\mu_0} 2g + \frac{B}{\mu_{Fe}} l_{core} = N_{id} \quad (40)$$

where l_{core} is the total length of flux routes in the steel core and μ_{PM} is the permeability of the PM. $\mu_{Fe} = 40000\mu_0$ for electrical steel. As a result, $\frac{B}{\mu_{Fe}} l_{core}$ is quite minimal and is thus often neglected. As $\mu_{PM} \approx \mu_0$, it can be concluded that

$$B = \frac{\mu_0 N}{2(g+h_m)} i_d \quad (41)$$

Considering the cross-sectional area of the air gap, $A \approx \frac{1}{2}\pi D_r l_{st}$ where D_r is the diameter and l_{st} is the stack length. Recognising $N\Phi = NB \times A = L_d i_d$, where N represent the d-axis windings number of turns. Thus d-axis inductance can be written as

$$L_d = \frac{\mu_0 N^2 A}{2(g+h_m)} \quad (42)$$

By applying Amper's law to Figure 8.1 (b) it can be obtained that $L_q = \frac{\mu_0 N^2 A}{2(g+h_m)}$ that is identical to that of L_d, hence L_d=L_q. In SPMSMs, the effective air gap is consistent across the rotor's perimeter, in other words there is no saliency.

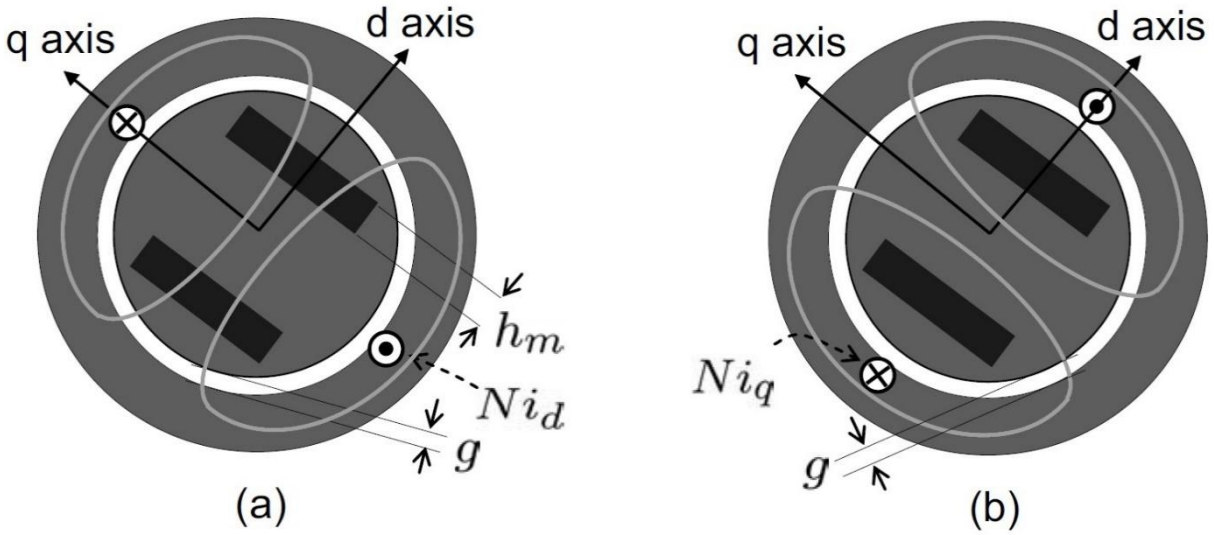

Figure 8. 2: (a) d–axis flux path, (b) q–axis flux path [33]

On the other hand, in IPMSMs, the reluctances vary based on the flux routes. According to Figure 8.2 (a) and (b), PM is along the d–axis flow, whereas none is observed along the q–axis flux. Consequently, the d–axis reluctance is larger than that of the q–axis, indicating that the d–axis inductance is less than that of the q–axis. The rotor saliency contributes to reluctance torque which improves the shaft torque. [33]

$$L_d = \frac{\mu_0 N^2 A}{2(g+h_m)} \quad (43)$$

$$L_q = \frac{\mu_0 N^2 A}{2g} \quad (44)$$

Appendix 2: Clarke and Park Transformation of Three-Phase Stator Winding Voltages

Taking the three-phase voltage signal from the inverter as a starting point. By using the previously explained axis transformation that is based on Clarke transformation, the three-phase voltages are transformed into stationary dq axis form, equation (45) and equation (46). [43]

$$V_{ds} = \frac{2}{3}V_a - \frac{1}{3}V_b - \frac{1}{3}V_c \quad (45)$$

$$V_{qs} = \frac{\sqrt{3}}{3}V_b - \frac{\sqrt{3}}{3}V_c \quad (46)$$

To transfer from stationary reference frame into rotary reference frame Park transformation is employed where Euler's formula is used to account for the changes related to the rotation of the shaft. Two-axis voltage values in rotary reference frame are demonstrated in equation (47) and equation (48).

$$V_{ds}^r = V_{ds}\cos\theta + V_{qs}\sin\theta \quad (47)$$

$$V_{qs}^r = -V_{ds}\sin\theta + V_{qs}\cos\theta \quad (48)$$

Appendix 3: SPM Machine Plate Drawing

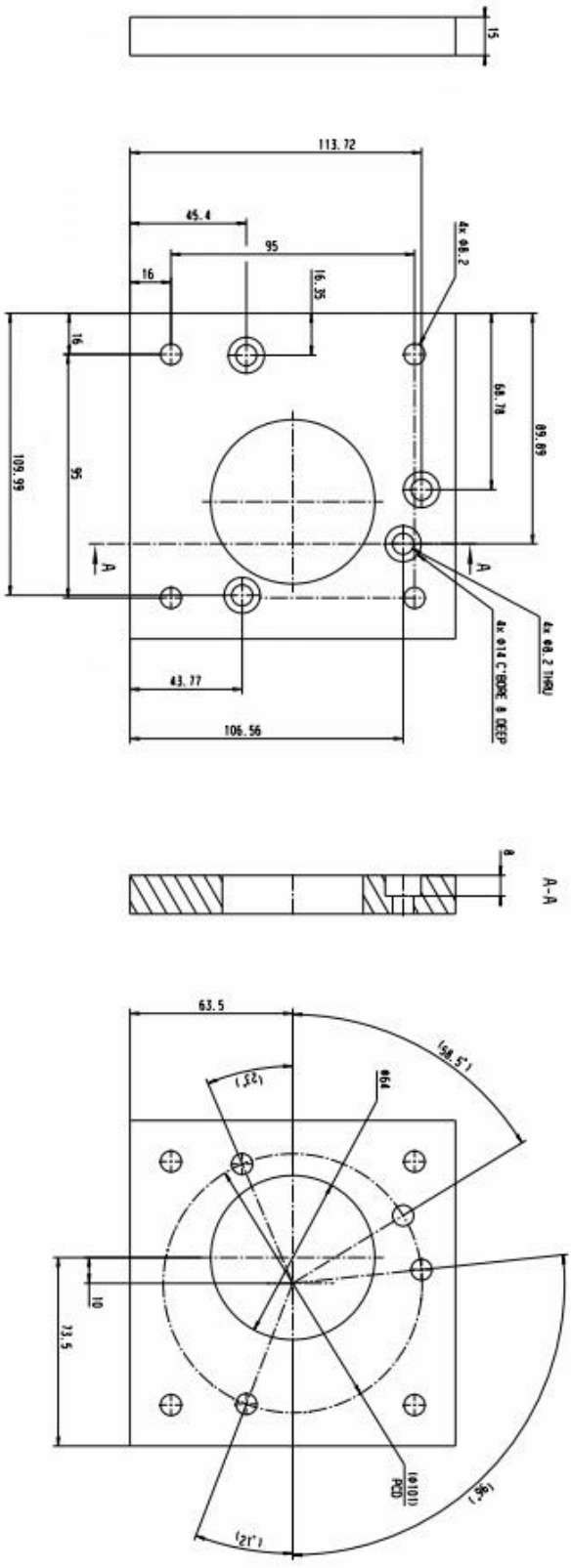

Figure 8. 3: SPM Machine plate drawing

Appendix 4: Component U_a And U_b Equation Derivation

$$\overline{V_R} sin\lambda = U_b sin60^o \quad (49)$$

$$U_b = \overline{V_R} \frac{sin\lambda}{sin60^o} = \frac{2}{\sqrt{3}} \overline{V_R} sin\lambda \quad (50)$$

$$U_a = \overline{V_R} cos\lambda = U_b cos60^o \quad (51)$$

$$= \overline{V_R} cos\lambda - \frac{2}{\sqrt{3}} \overline{V_R} sin\lambda . cos60^o \quad (52)$$

$$= \overline{V_R} \frac{2}{\sqrt{3}} \left\{ \frac{\sqrt{3}}{2} cos\lambda - sin\lambda . cos60^o \right\} \quad (53)$$

$$= \overline{V_R} \frac{2}{\sqrt{3}} \{ sin60^o cos\lambda - sin\lambda . cos60^o \} \quad (54)$$

$$U_a = \frac{2}{\sqrt{3}} |\overline{V_R}| \sin (60^o - \lambda) \quad (55)$$

$$U_b = \frac{2}{\sqrt{3}} |\overline{V_R}| \sin (\lambda) \quad (56)$$

Appendix.5: Space Vector Modulation Block

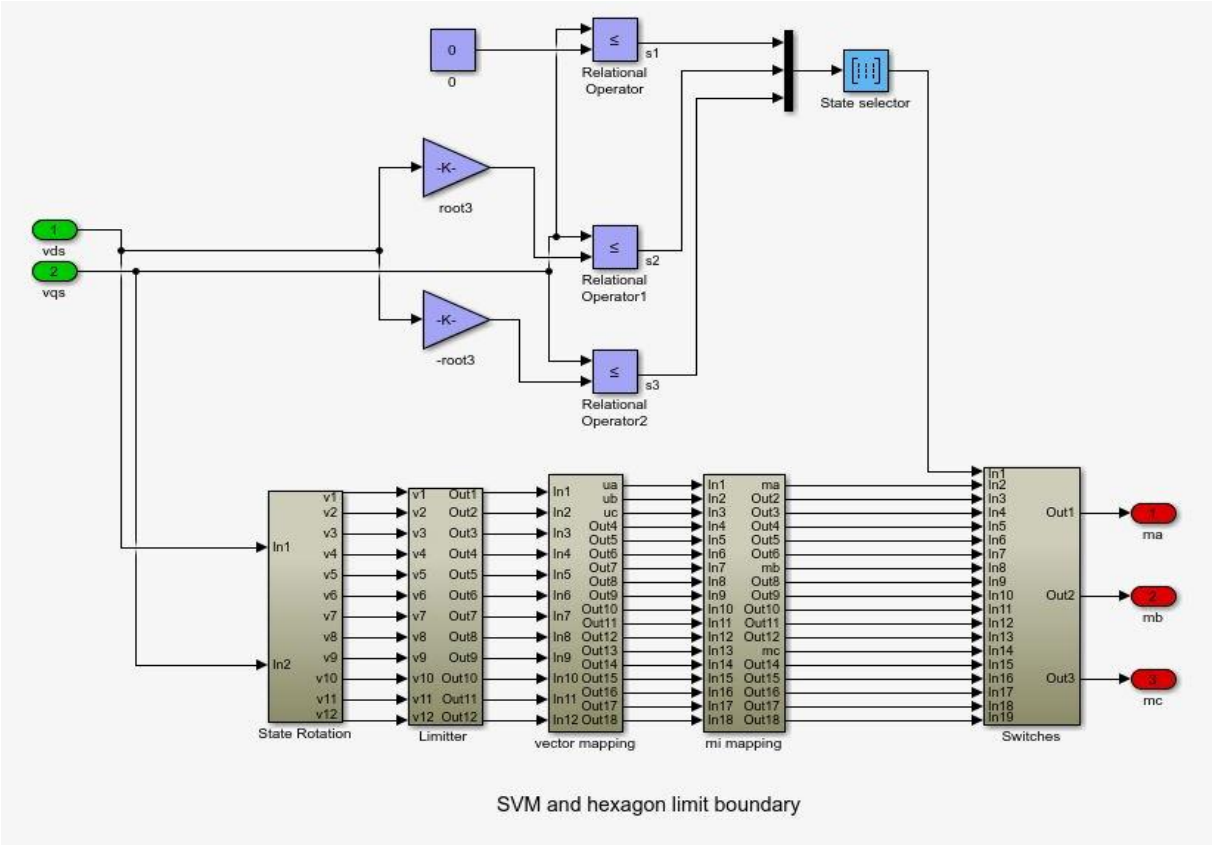

Figure 8. 4: Space Vector Modulation Block

Appendix 6: Oscilloscope Screenshot of Sinewave Torque

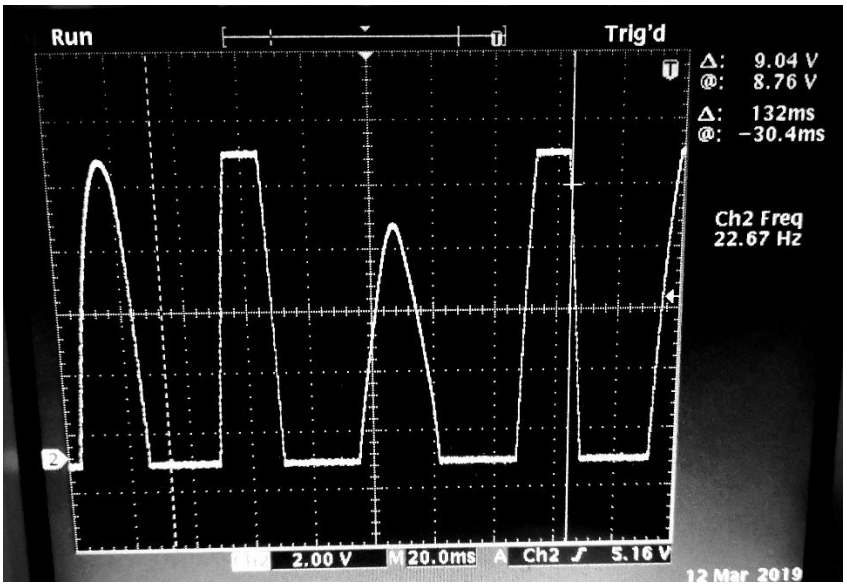

Figure 8. 5: Oscilloscope screenshot of sinewave torque

Appendix.7: Current Measurement Device

Figure 8. 6: Current measurement device

Figure 8. 7: Current Clamp

Appendix 8: Wireshark Graphical User Interface

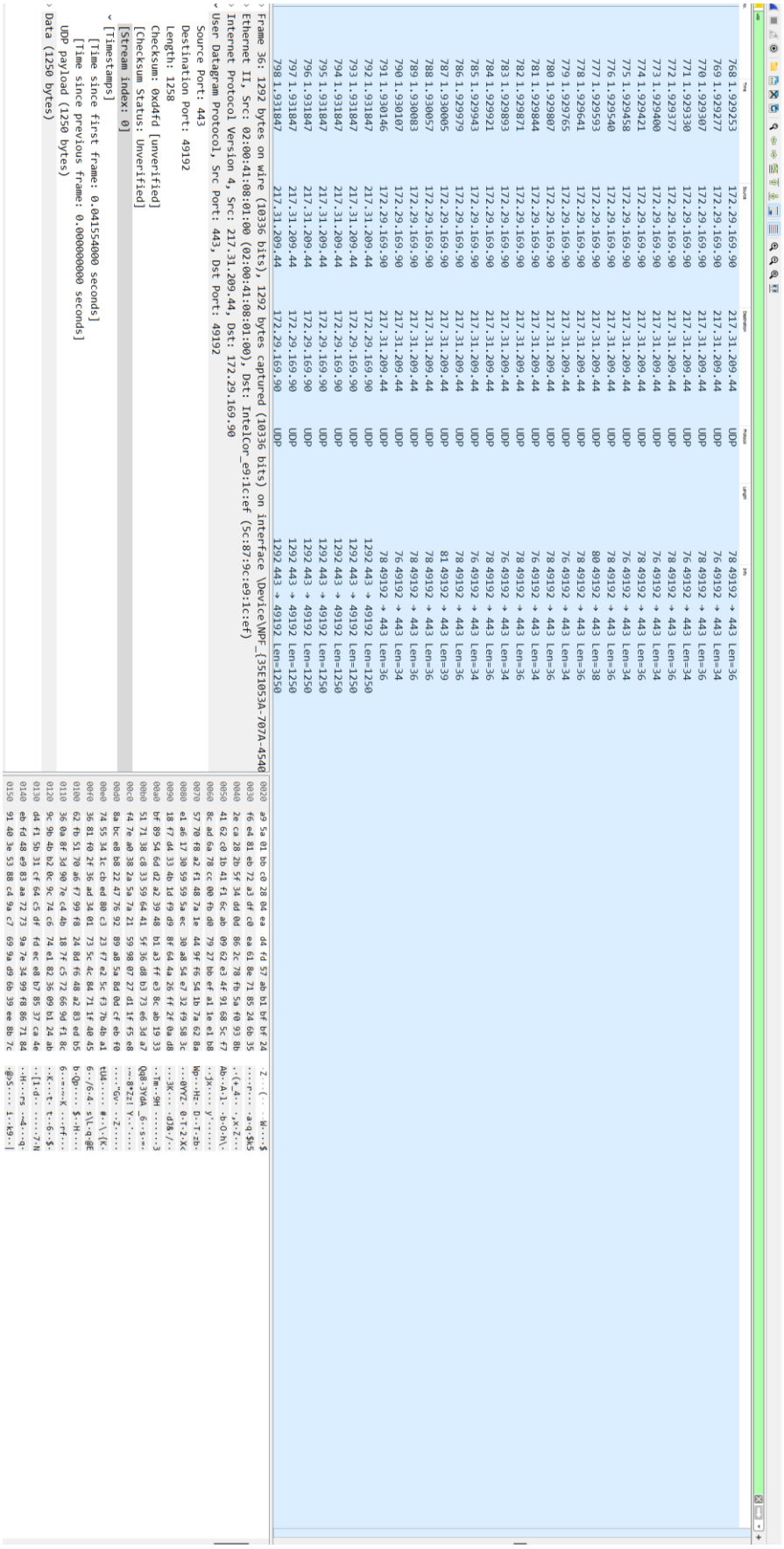

Figure 8. 8: Wireshark Graphical User Interface

Appendix.9 Packet Loss Simulink Block

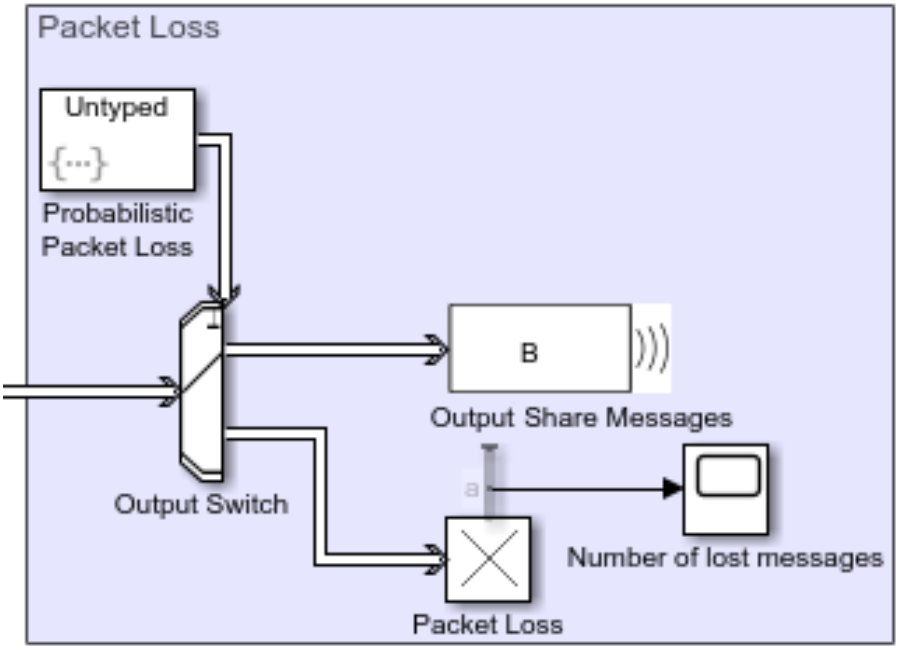

Figure 8. 9: Packet Loss Simulink Block [52]

Script to calculate probability:

```
persistent rngInit;
if isempty(rngInit)
    seed = 12345;
    rng(seed);
    rngInit = true;
end
% Pattern: Uniform distribution
% m: Minimum, M: Maximum
m = 0; M = 1;
x = m + (M - m) * rand;
% x is generated from uniform distribution and
% takes values between |0| and |1|.
if x > 0.1
   % Entity carries data |1| and this forces Output switch to select
   % output |1| to forward entities to receive components.
   entity  = 1;
else
   % Entity carries data |2| and this forces Output switch to select
   % output |2| and this represents a packet loss.
    entity = 1;
end
```

Appendix.10: Indmeas (Crane Electronics) Torque Sensor

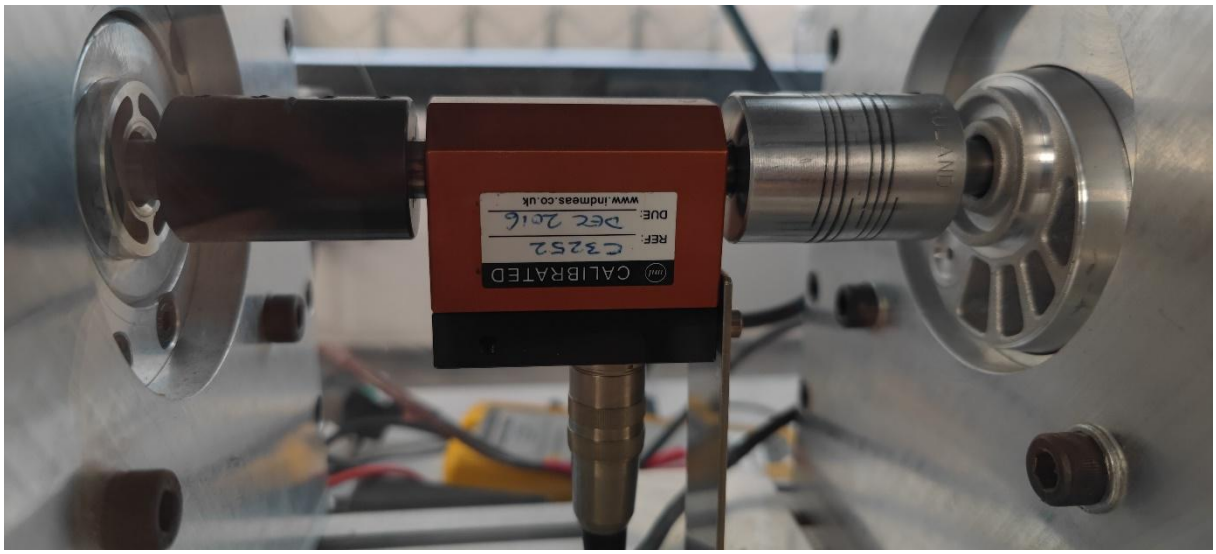

Figure 8. 10: Indmeas (Crane Electronics) Torque Sensor

Appendix 11: Current Signal Frequency Calculation

$$N = \frac{120 \times f}{P} \qquad (57)$$

N: Speed in rpm

F: Frequency

P: number of poles

For the motor used in this project and operating speed of 1020rpm, frequency is calculated to be 68Hz.

Appendix 12: dSpace Scalexio I/O Capabilities

I/O Board	Compatibility
DS2680 (I/O Unit)	Rack System
DS2690 (Digital I/O Board)	Rack System
DS2601 (Signal Measurement Board)	Rack System
DS2621 (Signal Generation Board)	Rack System
DS2642 (FIU & Power Switch Board)	Rack System
DS2671 (Bus Board)	Rack System
DS6101 (Multi-I/O Board)	Rack System and LabBox
DS6201 (Digital I/O Board)	Rack System and LabBox
DS6202 (Digital I/O Board)	Rack System and LabBox
DS6221 (A/D Board)	Rack System and LabBox
DS2655 (FPGA Base Board)	Rack System and LabBox
DS6301 (CAN/LIN Board)	Rack System and LabBox
DS6311 (Flex Ray Board)	Rack System and LabBox
DS6331 (Ethernet Board)	Rack System and LabBox
DS6341 (CAN Board)	Rack System and LabBox
DS6351 (LIN Board)	Rack System and LabBox

Table 8. 1: dSpace Scalexio I/O board list

Appendix.13: Axis Transformation

Considering the three-phase stator windings, Figure 8.11 depicts the mmf, magnetomotive force, and the sinusoidal distribution of mmf around the airgap. Each coil produces an mmf and the three coils are 120 degrees apart, hence the total stator current can be calculated from equation (58) that is a vector quantity. [40]

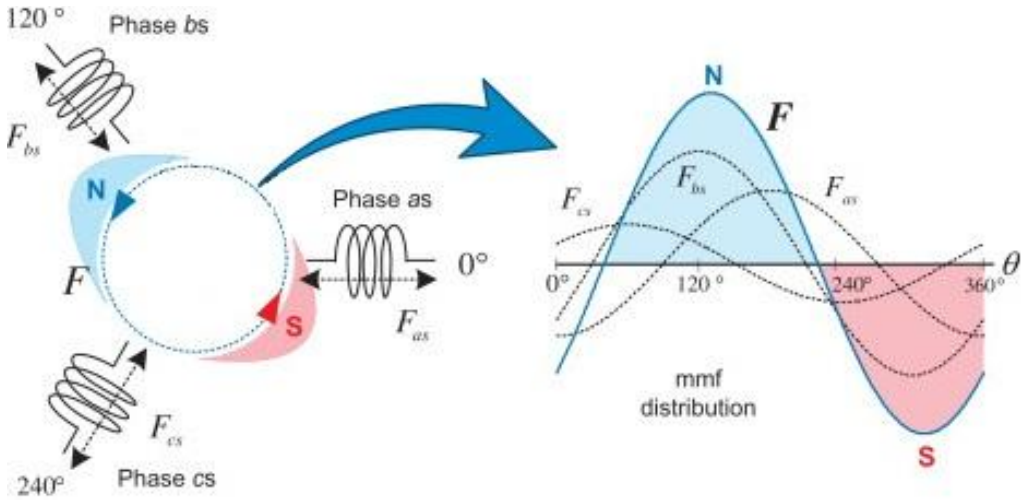

Figure 8. 11: mmf distribution

$$\overline{i_s} = \frac{2}{3}(\overline{i_a} + \overline{i_b} + \overline{i_c}) \qquad (58)$$

As the machine is a three-phase machine using a 120-degree operator, *a*, can be effective. The three-phase stator currents can be transformed into equation (59), equation (60) and equation (61). Also, total stator current can be rewritten as equation (62). [35]

$$\overline{i_a} = i_a \exp(j0^{\circ}) = i_a \qquad (59)$$

$$\overline{i_b} = i_b \exp(j120^{\circ}) = a.i_b \qquad (60)$$

$$\overline{i_c} = i_c \exp(j240^{\circ}) = a^2.i_c \qquad (61)$$

$$\overline{i_s} = \frac{2}{3}(i_a + a.i_b + a^2.i_c) \qquad (62)$$

Taking the *a*-operator in rectangular coordinates

$$a = \exp(j120^{\circ}) = \cos(120^{\circ}) + jsin(120^{\circ}) = -\frac{1}{2} + j\frac{\sqrt{3}}{2} \qquad (63)$$

$$a = \exp(j240^{\circ}) = \cos(240^{\circ}) + jsin(240^{\circ}) = -\frac{1}{2} - j\frac{\sqrt{3}}{2} \qquad (64)$$

Transforming the three-phase stator currents into an equivalent two-phase frame. By substituting the rectangular form of a-operator in equation (65), the two-axis currents in term of the original three-phase currents can be found by separating the imaginary and real parts, equation (67) and equation (68).

$$\overline{i_s} = \frac{2}{3}(i_a + a.i_b + a^2.i_c) = i_{ds} + ji_{qs} \quad (65)$$

$$\frac{2}{3}\left\{i_a + -\frac{1}{2}i_b + j\frac{\sqrt{3}}{2}i_b - \frac{1}{2}i_c - j\frac{\sqrt{3}}{2}i_c\right\} = i_{ds} + ji_{qs} \quad (66)$$

$$i_{ds} = \frac{2}{3}i_a - \frac{1}{3}i_b - \frac{1}{3}i_c \quad (67)$$

$$i_{qs} = \frac{1}{\sqrt{3}}(i_b - i_c) \quad (68)$$

Therefore, transformation from three to two-axis can be accomplished by using equation (69), equation (70) and equation (71).

$$i_a = i_{ds} \quad (69)$$

$$i_b = -\frac{1}{2}i_{ds} + \frac{\sqrt{3}}{2}i_{qs} \quad (70)$$

$$i_c = -\frac{1}{2}i_{ds} - \frac{\sqrt{3}}{2}i_{qs} \quad (71)$$

This approach is later used to transform the three-phase voltage and flux linkage into two-phase frame.

Appendix 14: The Cross-Section of The Most Prevalent IPM And SPM Topologies

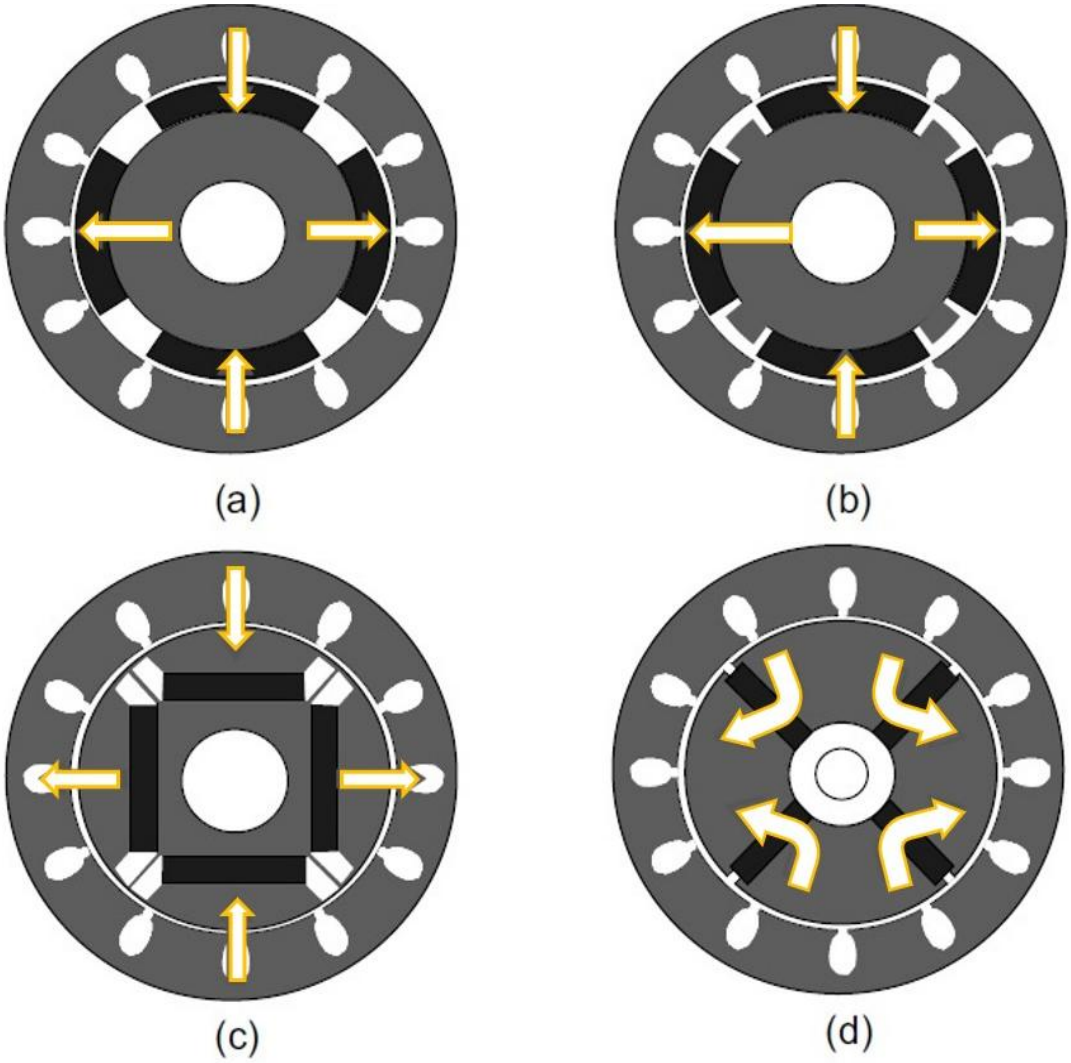

Figure 8. 12: a. Surface Mounted b. Inset c. Interior d. Flux Concentration Interior [33]

Figure 8.12 depicts cross-sectional views of surface-mounted (a and b) and interior PMSMs (c and d) consist of four poles shown in dark. The position of the magnets influences the flux linkage and overall performance of the machine. For instance, In Figure 8.12 (b), permanent magnets (PMs) are placed into the groove of the rotor surface to form an inset magnet motor. Even though the magnets are on the surface, the inset magnet motor has a distinct magnetic saliency, like IPMSMs. In particular, q–axis inductance is greater than d–axis inductance. The flux concentration configuration is shown in Figure 8.12 (d), in which the air-gap flux density may be raised above the residual PM flux density. It is important to note that no ferromagnetic materials are employed in the region around the shaft to impede flux passage through the central portion. Appendix.1 explain the PMSM inductance in greater detail. [35, 41]

Appendix 15: Phase Connection Pins of The Test Machine

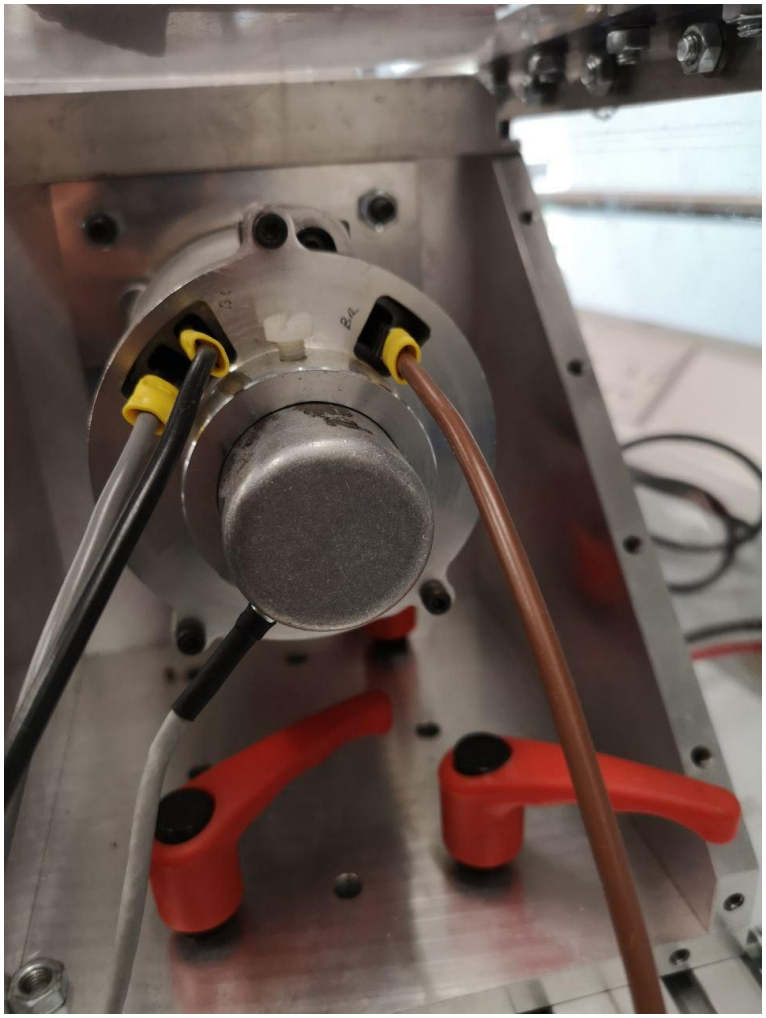

Figure 8. 13: SPM under test line-line pins

Appendix.16: GUI for F28377D

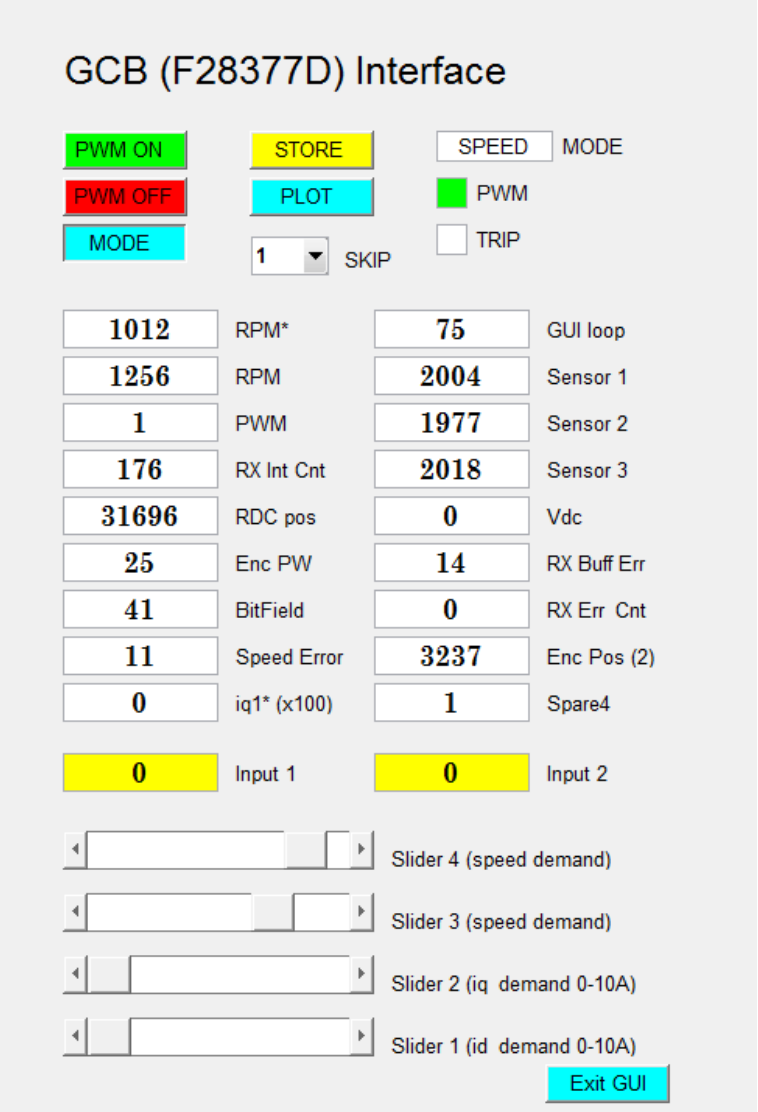

Figure 8. 14: GUI for F28377D

Appendix. 17: ODE45 and ODE8 Slope calculations

- **ODE45 (Runge-Kutta):**

$$k_1 = f(t_n, y_n) \quad (72)$$

$$k_2 = f(t_n + \frac{h}{2}, y_n + \frac{h}{2}k_1) \quad (73)$$

$$k_3 = f(t_n + \frac{h}{2}, y_n + \frac{h}{2}k_2) \quad (74)$$

$$k_4 = f(t_n + h, y_n + hk_2) \quad (75)$$

- **ODE8 (Dormand-Prince)**

$$k_i = f(t_n + c_i h, y_n + h\sum_{j=1}^{i-1} a_{ij}k_j) \quad (76)$$

Where a_{ij}, b_i and c_i are coefficients specific to the particular Runge-Kutta method (Dormand-Prince for this solver)

Appendix 18: Additional Illustrations for Simulink Model Blocks and Results

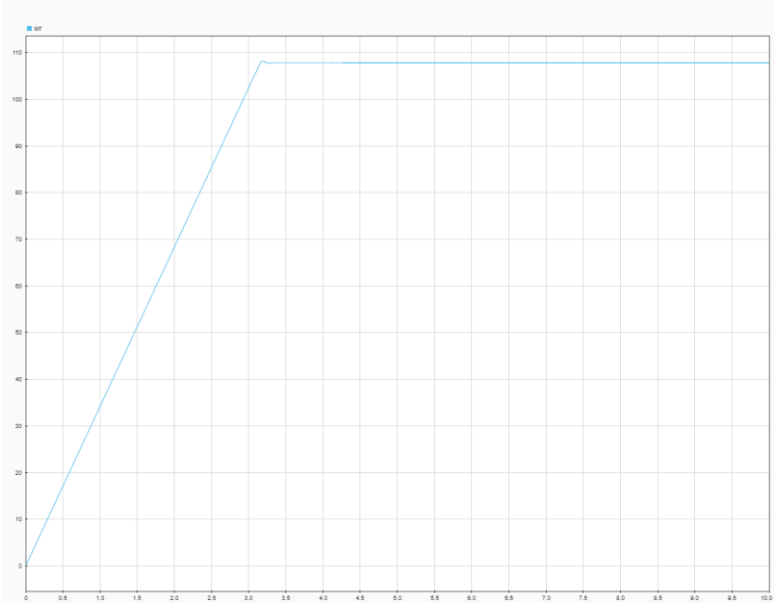

Figure 8. 15: Simulink Model Speed

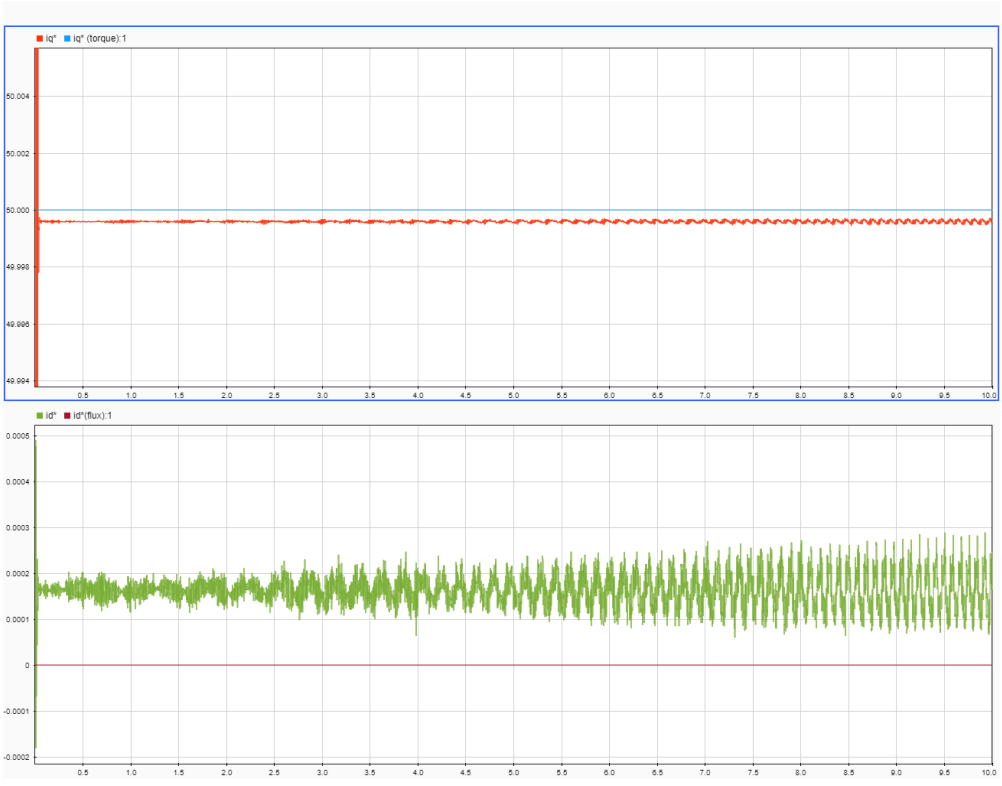

Figure 8. 16: iq and id demand vs achieved iq and id

Appendix 19: PrefSONAR Screenshots

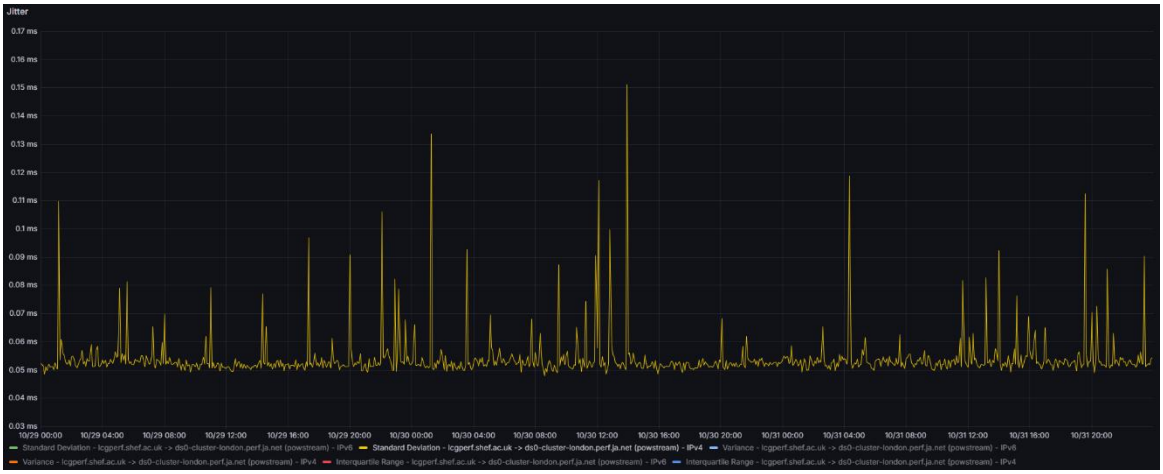

Figure 8. 17: PrefSonar jitter data [49]

Figure 8. 18: PrefSonar latency data [49]

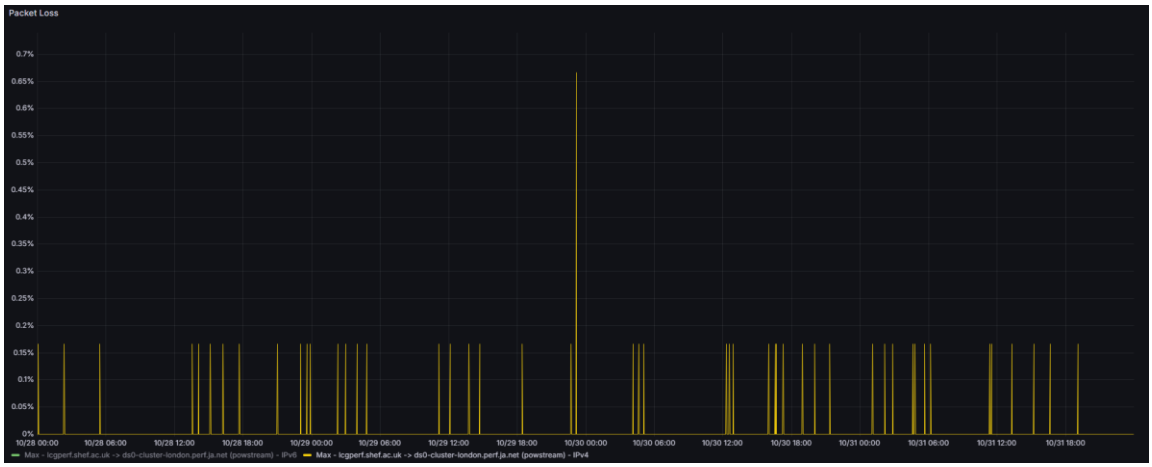

Figure 8. 19: PrefSonar packet loss data [49]

Appendix

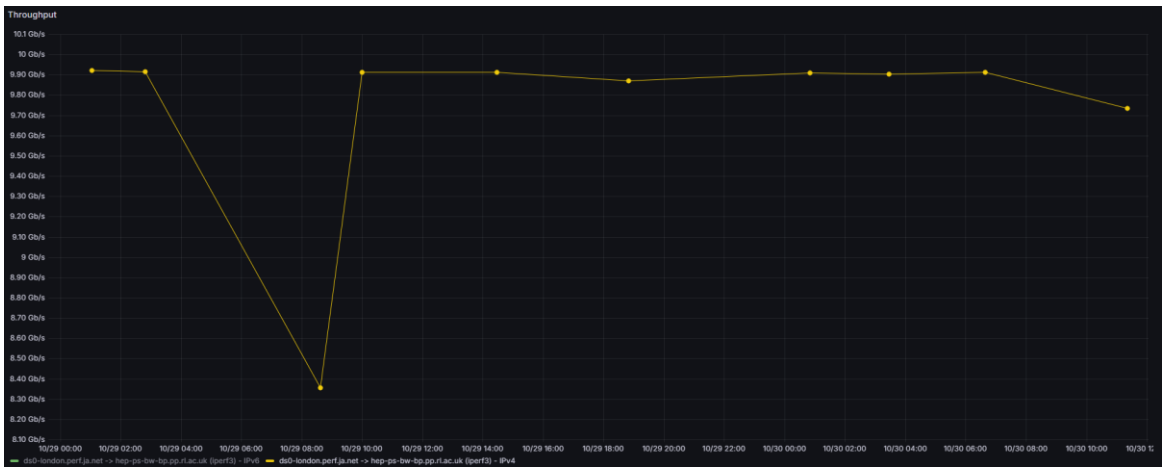

Figure 8. 20: PrefSonar throughput [49]

Appendix 20: ControlDesk Screenshots

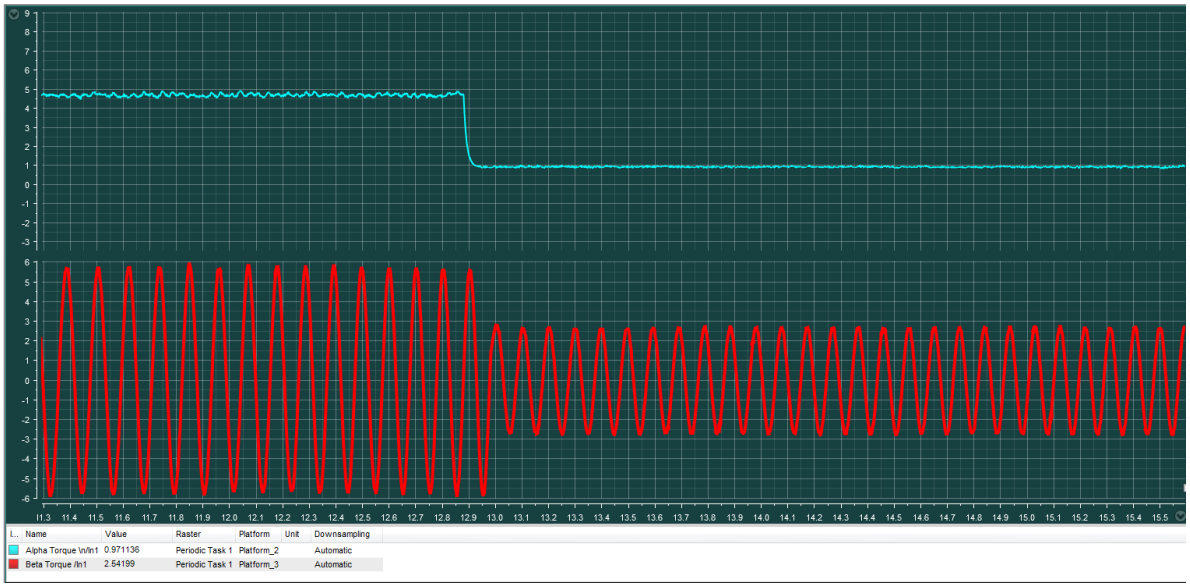

Figure 8. 21: Packet loss test

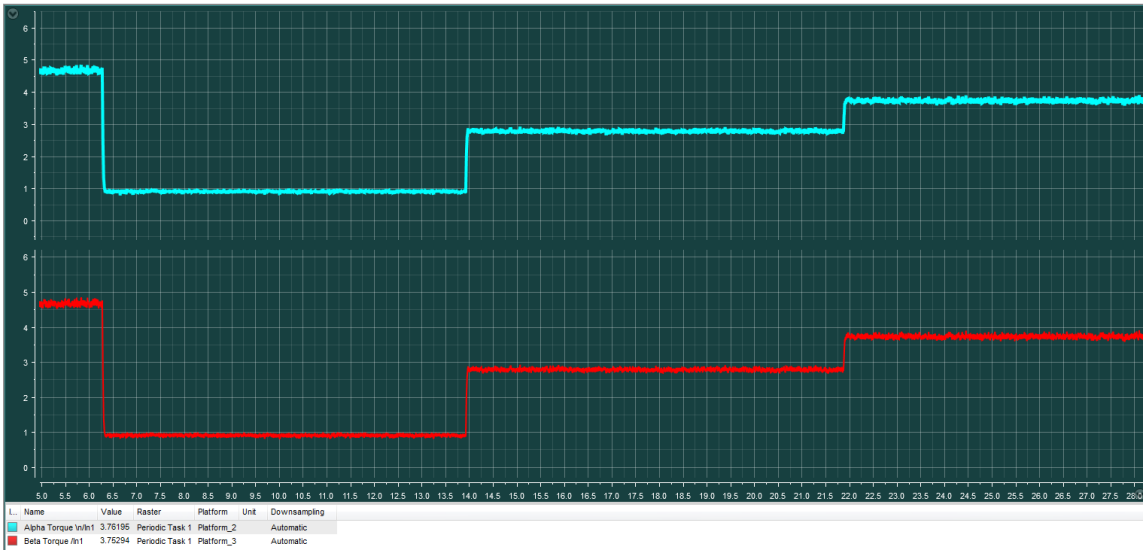

Figure 8. 22: Slave Torque (Red) Master Torque (Green) at 20% Packet loss, flow chart implemented (ControlDesk)

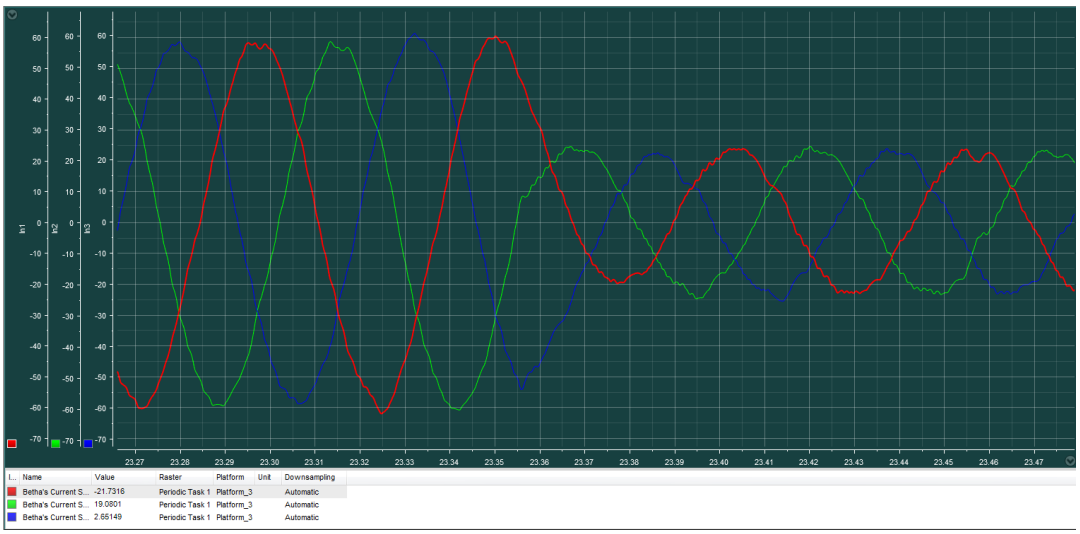

Figure 8. 23: Mixed local and obtained Current Signal (ControlDesk)

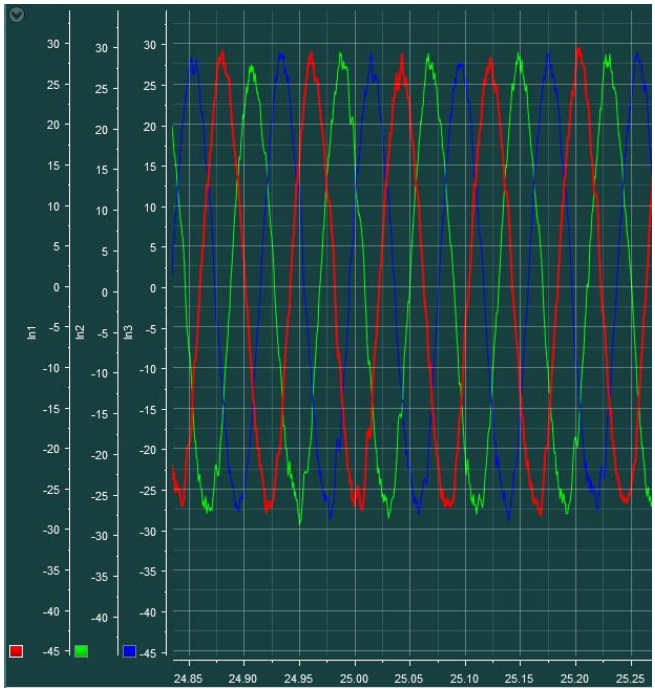

Figure 8. 24: Three-Phase Current Signal After processing the packet compensation blocks.

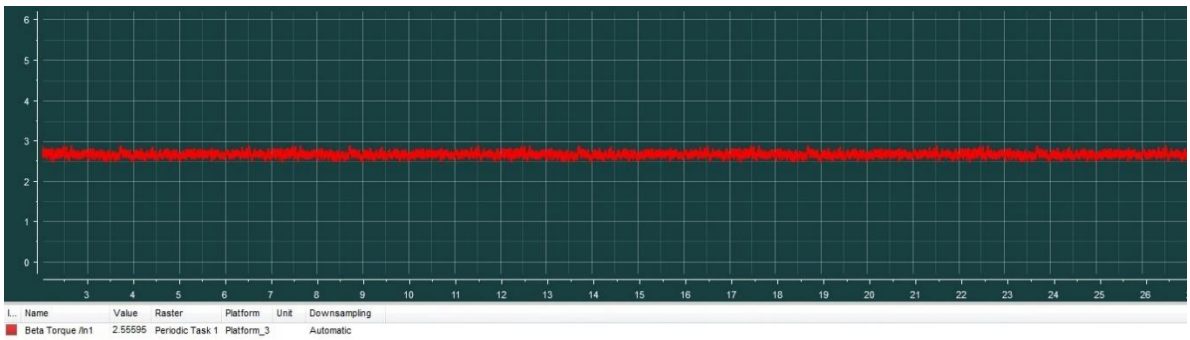

Figure 8. 25: Torque Signal After processing the packet compensation blocks.

Appendix 21: Simulink Blocks Screenshots

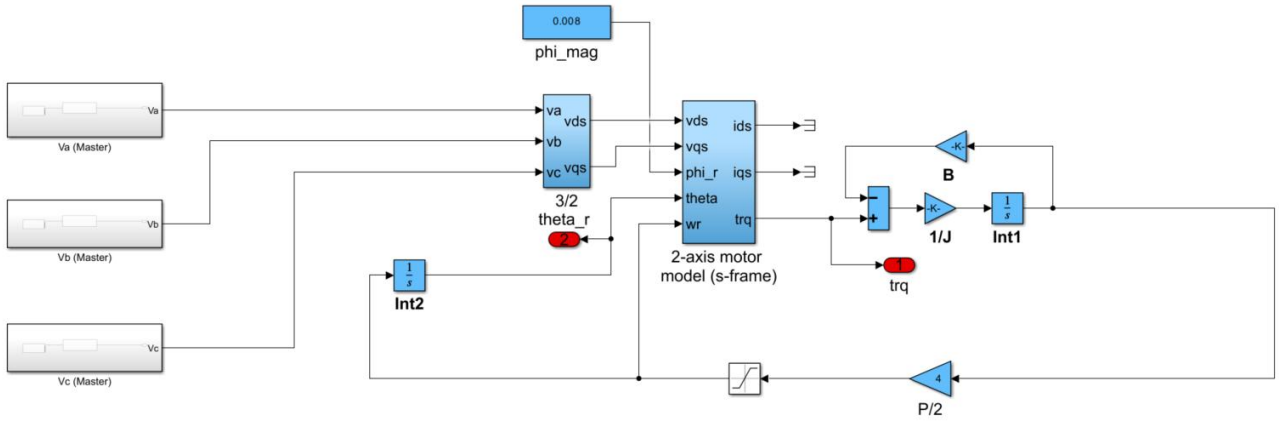

Figure 8. 26: Slave model

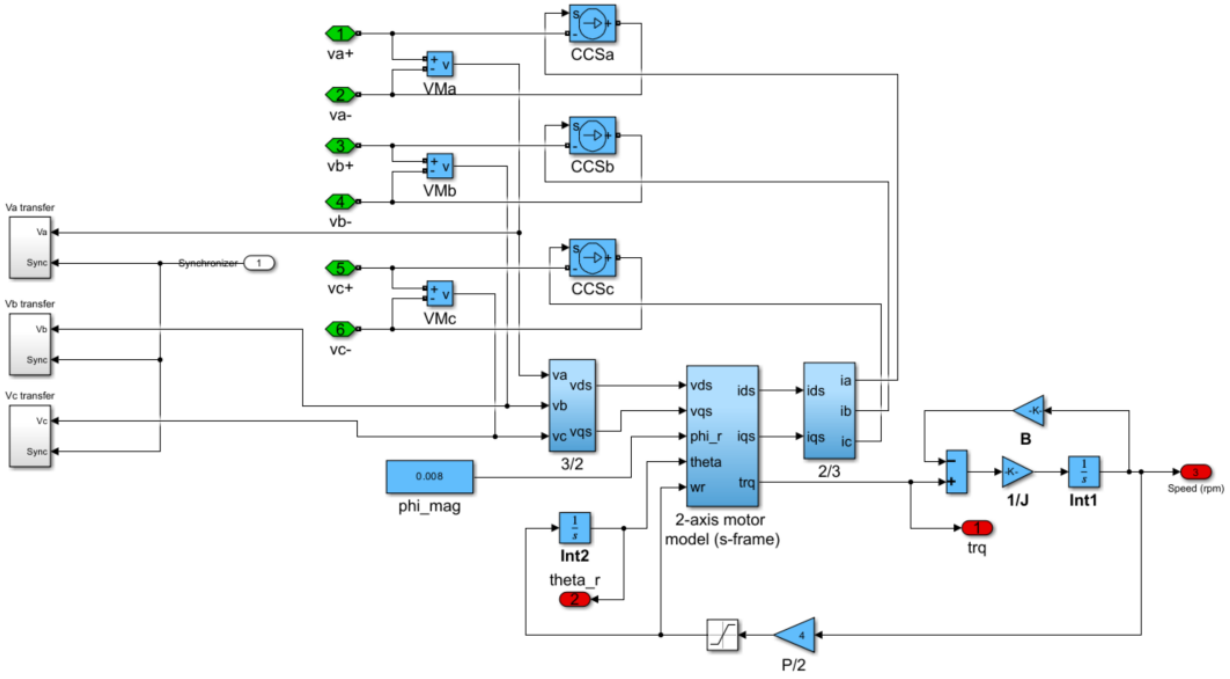

Figure 8. 27: UDP Blocks Within the Master Model

Appendix 22: Network Impairment Blocks

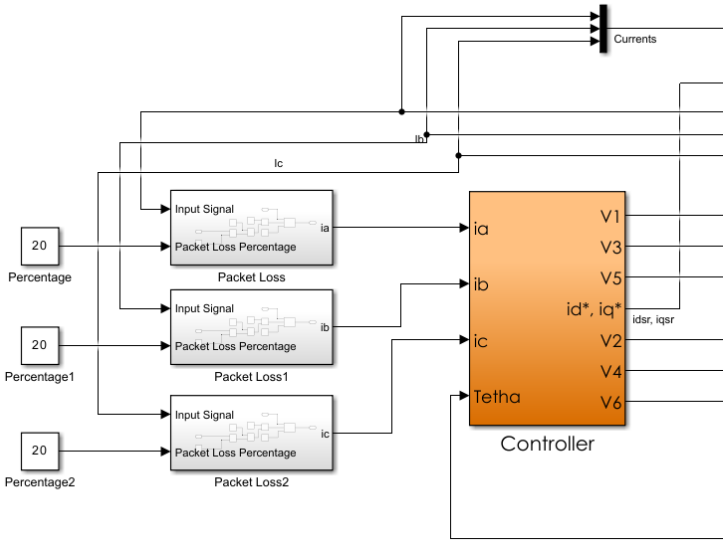

Figure 5. 23: Packet loss block placement in the master models

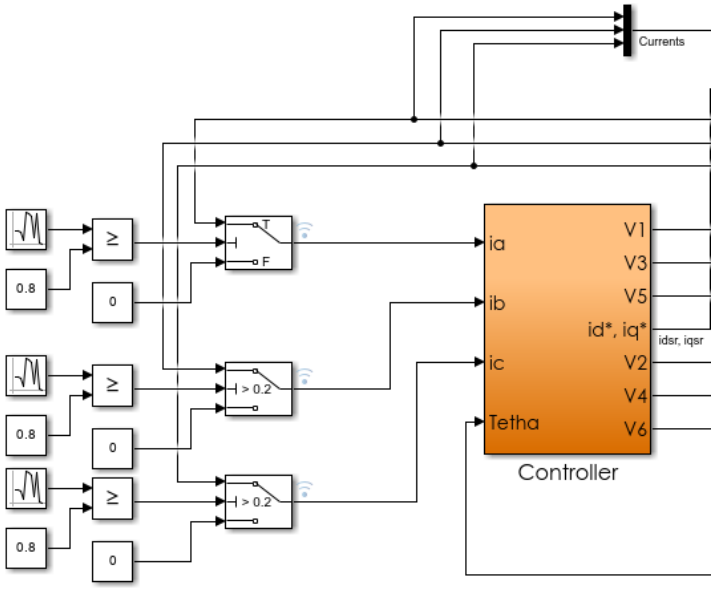

Figure 5. 24: Jitter blocks placement in the models

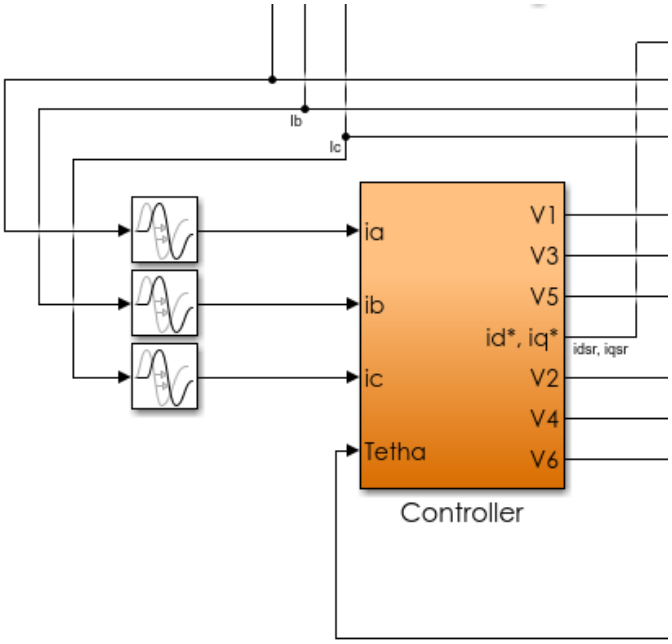

Figure 5. 25: Latency blocks placement in the master model