ESTIMATION OF STATOR RESISTANCE OF INDUCTION MOTOR USING INNOVATION BASED ADAPTIVE EXTENDED KALMAN FILTER

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES
OF
MIDDLE EAST TECHNICAL UNIVERSITY

BY

MEHMET ZEKI YIRTAR

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR
THE DEGREE OF MASTER OF SCIENCE
IN
ELECTRICAL AND ELECTRONICS ENGINEERING

SEPTEMBER 2021

Approval of the thesis:

ESTIMATION OF STATOR RESISTANCE OF INDUCTION MOTOR USING INNOVATION BASED ADAPTIVE EXTENDED KALMAN FILTER

submitted by **MEHMET ZEKI YIRTAR** in partial fulfillment of the requirements for the degree of **Master of Science in Electrical and Electronics Engineering Department, Middle East Technical University** by,

Prof. Dr. Halil Kalıpçılar
Dean, Graduate School of **Natural and Applied Sciences** _____

Prof. Dr. İlkay Ulusoy
Head of Department, **Electrical and Electronics Engineering** _____

Assist. Prof. Dr. Emine Bostancı Özkan
Supervisor, **Electrical and Electronics Engineering, METU** _____

Prof. Dr. H. Bülent Ertan
Co-supervisor, **Electrical and Electronics Engineering, Atılım University** _____

Examining Committee Members:

Prof. Dr. Muammer Ermiş
Electrical and Electronics Engineering, METU _____

Assist. Prof. Dr. Emine Bostancı Özkan
Electrical and Electronics Engineering, METU _____

Prof. Dr. Kemal Leblebicioğlu
Electrical and Electronics Engineering, METU _____

Assoc. Prof. Dr. Murat Göl
Electrical and Electronics Engineering, METU _____

Assist. Prof. Dr. Remzi İnan
Electrical and Electronics Engineering, ISUBU _____

Date:24.09.2021

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Surname: Mehmet Zeki YIRTAR

Signature :

ABSTRACT

ESTIMATION OF STATOR RESISTANCE OF INDUCTION MOTOR USING INNOVATION BASED ADAPTIVE EXTENDED KALMAN FILTER

YIRTAR, Mehmet Zeki
M.S., Department of Electrical and Electronics Engineering
Supervisor: Assist. Prof. Dr. Emine Bostancı Özkan
Co-Supervisor: Prof. Dr. H. Bülent Ertan

September 2021, 118 pages

In the industry, induction motors are widely utilized due to their low cost and low maintenance requirements. To get better performance from the motor, vector control (FOC) is introduced in the last decades. In this thesis, the software and hardware parameters that affect the performance of the vector-controlled induction motor will be investigated by simulations and experiments. PI controller design, axis decoupling, dc-link compensation and stator resistor estimation are implemented and simulated in Simulink then verified in an experimental setup. Having an extra sensor to sense the rotor speed brings an additional cost to the system; thus, sensorless vector control is widely used to control induction motors. Sensorless vector control generally uses the induction motor's voltage model to estimate the rotor flux; to accurately estimate the flux, the information of stator resistance value is crucial. An Innovation based adaptive extended Kalman filter model is implemented in the real-time environment to estimate the stator resistance of the motor.

The experimental setup is controlled by DSPACE DS1104, which is programmed by the Simulink coder. The proposed algorithms are implemented and tested in Simulink

first; after verification in the simulation environment, the algorithm is embedded into Dspace DS1104, and experiments are conducted.

Keywords: Vector Control, Stator Resistance Estimation, Innovation Based Adaptive Extended Kalman Filter

ÖZ

YENİLİK TEMELLİ UYARLANABİLİR GENİŞLETİLMİŞ KALMAN FİLTRESİ KULLANILARAK ASENKRON MOTORUN STATOR DİRENCİ KESTİRİMİ

YIRTAR, Mehmet Zeki
Yüksek Lisans, Elektrik ve Elektronik Mühendisliği Bölümü
Tez Yöneticisi: Dr. Öğr. Üyesi. Emine Bostancı Özkan
Ortak Tez Yöneticisi: Prof. Dr. H. Bülent Ertan

Eylül 2021 , 118 sayfa

Asenkron motorlar düşük maliyet ve düşük bakım ihtiyaçlarından dolayı endüstride oldukça yaygın kullanılırlar. Asenkron motoru etkin bir şekilde kullanabilmek için son yıllarda vektör kontrol kullanılmaya başlanmıştır. Bu tezde vektör kontrol ile kontrol edilen asenkron motorun performansını etkileyen yazılımsal ve donanımsal parametreler simülasyonlar ve deneylerle incelenecektir. PI kontrolör tasarımı, eksen ayırma, DC-link kompanzasyonu ve stator direnç tahmini Simulink'te gerçekleştirilmiş ve simüle edilmiş ardından deneysel bir kurulumda doğrulanmıştır. Hız sensörü kullanmak sisteme fazladan maliyet getirir, bu sebepten dolayı asenkron motorlar sensörsüz vektör kontrolü ile yaygın olarak sürülür. Sensörsüz vektör kontrolünde rotor akı pozisyonunun doğru tahmin edilebilmesi için genelde motorun voltaj modeli kullanılır, akı pozisyonunun doğru tahmin edilebilmesi için stator direncinin doğru bilinmesi önemlidir. Stator direncinin ölçümü için yenilik temelli uyarlanabilir genişletilmiş Kalman filtresi gerçek zamanlı ortamda kullanılıp deneyler yapılmıştır.

Deney düzeneği, Simulink kodlayıcı tarafından programlanan Dspace DS1104 tarafından kontrol edilir. Önerilen algoritmalar önce Simulink'te gerçeklendi ve test edildi; simülasyon ortamında doğrulama yapıldıktan sonra, algoritmalar Dspace DS1104'e gömüldü ve deneyler yapıldı.

Anahtar Kelimeler: Vektör Kontrol, Stator Direnç Kestirimi, Yenilik Temelli Uyarlanabilir Genişletilmiş Kalman Filtresi

To my family

ACKNOWLEDGMENTS

I would like to express my sincere gratitude to my supervisor Assist. Prof. Dr. Emine BOSTANCI ÖZKAN and my co-advisor Prof. Dr. H. Bülent ERTAN for constantly assisting and motivating me. They have been excellent mentors in both academic and real-life circumstances. Their unwavering support and advice, along with their expertise, have enabled me to complete my thesis work.

I am very grateful to TÜBİTAK-SAGE for giving me the opportunity to complete this study.

I would like to express my gratitude to Assist. Prof.Dr. Remzi Inan for his assistance during the implementation of the IAEKF algorithm and for always being available for technical discussion and help.

For the feedback that contributed to my thesis, I am grateful to the members of my thesis committee, namely Prof. Dr. Muammer ERMİŞ, Prof. Dr. Kemal LEBLECİ-COĞLU, and Assoc. Prof. Dr. Murat GÖL.

METU made distinctive contributions to my problem-solving skills, personality, critical thinking, and work ethic, among other things. I would like to express my sincere gratitude to all of my professors at METU for sharing their rich knowledge and expertise.

I would like to express my gratitude to my family for their unwavering love and support. Without their encouragement, support and patience I would not have finished this thesis work.

TABLE OF CONTENTS

ABSTRACT v

ÖZ vii

ACKNOWLEDGMENTS x

TABLE OF CONTENTS xi

LIST OF TABLES xv

LIST OF FIGURES xvii

LIST OF ABBREVIATIONS xxiii

CHAPTERS

1 INTRODUCTION 1

1.1 Electric Motors and Induction Motors 1

1.2 Vector Control 2

1.3 Parameters and States That are Effective for High Dynamic Performance 4

1.4 Motor Parameter Estimation 5

1.5 The Outline of the Thesis 9

2 THE HARDWARE AND THE SIMULATION ENVIRONMENT FOR VECTOR CONTROL 11

2.1 Introduction 11

2.2 Brief Explanation of the Hardware 11

2.2.1 Current and Voltage Sense Amplifier Circuits 13
2.2.2 Power Stage and Motor 16
2.3 Vector Control 17
2.3.1 Vector Control of Induction Motor 18
2.4 Software Environment 20
2.4.1 Variable Definitions and Initialization Module 22
2.4.2 Obtaining the Rated Flux 22
2.4.3 Voltage and Current Measurement 23
2.4.4 Clarke Transformation Module 23
2.4.5 Park Transformation Module 25
2.4.6 PI Controller Module 26
2.4.7 Anti-Windup Mechanism 29
2.4.8 Axis Decoupling 30
2.4.9 DQ Voltage Limiter Module 31
2.4.10 Inverse Park Transformation Module 32
2.4.11 Rotor Flux Angle Estimation Module 32
2.4.12 Dead Time Compensation 35
2.4.13 Resistive Braking Module 35
2.4.14 Space Vector PWM Module 36
2.4.15 DC Link Compensation 40
2.4.16 Speed Sensing 41
2.4.17 Speed Loop 44
2.4.18 PWM Generation and Dead Time Insertion Module 44

2.4.19 Torque Calculation Module 45

2.5 Testing the Influence of the Series PI Controller on the Torque Producing Current $\mathbf{I_{qs}}$ 46

2.6 Verification of the Model 48

2.7 Conclusion 50

3 REAL-TIME STATOR RESISTANCE ESTIMATION WITH INNOVATION BASED ADAPTIVE EXTENDED KALMAN FILTER 51

3.1 Introduction 51

3.1.1 The Kalman Filter 52

3.1.2 IAEKF Design 61

3.1.2.1 Calculation of the Measurement Covariance Matrix . . . 62

3.1.3 Conclusion 64

4 SIMULATION MODEL OF THE IEAKF ALGORITHM 65

4.1 Simulink Model of the IAEKF 65

4.2 Simulation Results 69

4.2.1 Stator Resistance Estimation for Standstill Condition 70

4.2.2 Stator Resistance Estimation for Running Condition 72

4.3 Conclusion 73

5 EXPERIMENT RESULTS FOR STANDSTILL TESTS 75

5.1 Introduction 75

5.1.1 Experimental Procedure 76

5.1.1.1 Standstill Test Conducted with Variable Voltage Transformer 83

5.1.1.2 Standstill Test Conducted with PWM Inverter Driven at 10Hz 84

5.1.1.3 Standstill Test Conducted with PWM Inverter Driven at 25Hz . . . 86

5.1.1.4 Standstill Test Conducted with PWM Inverter Driven at 50Hz . . . 87

5.1.1.5 Standstill Test Conducted with PWM Inverter Driven at 75Hz . . . 88

5.1.1.6 Standstill Test Conducted with PWM Inverter Driven at 100Hz . . . 90

5.2 Conclusion . . . 92

6 EXPERIMENT RESULTS FOR RUNNING TESTS . . . 93

6.1 Introduction . . . 93

6.2 Experimental Procedure . . . 93

6.2.1 $\mathbf{R_s}$ Estimation at 600 RPM . . . 94

6.2.2 $\mathbf{R_s}$ Estimation at 1500 RPM . . . 95

6.2.3 $\mathbf{R_s}$ Estimation at 3000 RPM . . . 97

6.2.4 $\mathbf{R_s}$ Value vs Speed Estimation Performance of Sensorless Vector-Controlled Induction Motor . . . 98

6.2.5 Stator Resistance Estimation Under Load Torque . . . 99

6.2.6 The Experimental Setup and the DSPACE ControlDesk Software 100

6.3 Conclusion . . . 101

7 CONCLUSION . . . 103

REFERENCES . . . 105

APPENDICES

A EXPERIMENTAL MOTOR PARAMETERS . . . 111

B VECTOR CONTROL SOFTWARE . . . 113

LIST OF TABLES

TABLES

Table 2.1 Discretization methods to implement the integration [1] 29

Table 2.2 Switch states and phase to neutral and line-line voltages 39

Table 2.3 Vector-controlled induction motor's no-load test at 3000 RPM done in simulation and real-time environment 49

Table 2.4 Vector controlled induction motor's 3Nm loaded test at 3000 RPM done in simulation and real-time environment 49

Table 4.1 Estimated $\hat{R}_s$ for different elements of noise covariance matrix $R_k[A]$ values 69

Table 4.2 Estimated $\hat{R}_s$ for different window length N values 69

Table 5.1 Estimated $\hat{R}_s$ and percentage error for variable voltage transformer and PWM supply with different frequencies are tested, window length N=4 and $R_k[A^2] = 0.000459$ 78

Table 5.2 Estimated $\hat{R}_s$ and percentage error for variable voltage transformer and PWM supply with different frequencies are tested while series 1 Ω resistance is added, window length N=4 and $R_k[A^2] = 0.000459$ 78

Table 5.3 Estimated $\hat{R}_s$ value for various window length N values at 50Hz while $R_k[A^2] = 0.000459$ 79

Table 5.4 Estimated $\hat{R}_s$ value for various $R_k[A]$ values at 50 Hz while the window length is N=4 79

Table 6.1 Estimated $\hat{R}_s$ value at different speeds, window length N=4 and $R_k[A^2] = 1$ 94

Table 6.2 Estimated Speed value vs R_s value at different speeds 99

Table 6.3 Estimated speed error vs R_s at different speeds 99

Table A.1 Nameplate of the SE 2662-5H 111

Table A.2 Parameters of the SE 2662-5H 111

LIST OF FIGURES

FIGURES

Figure 1.1 Block diagram of motor drives 2

Figure 1.2 Vector control methods 3

Figure 1.3 Reference fluxes available for vector control 3

Figure 1.4 Vector control categorized according to position or speed sensor availability 4

Figure 1.5 Actual vs commanded rotor flux in parameter mismatch situation [2] 5

Figure 1.6 Parameter estimation 6

Figure 1.7 Stator resistor estimation techniques 7

Figure 2.1 Block diagram of the hardware used in this thesis 12

Figure 2.2 Current sense amplifier circuit's schematic diagram 13

Figure 2.3 Frequency response of the current sense amplifier schematic diagram given in Figure 2.2 14

Figure 2.4 Experimental setup of frequency response analyze tests 15

Figure 2.5 Transfer function of the current sensor filter circuit; blue: gain, red: phase 15

Figure 2.6 Motor drive hardware [3] 16

Figure 2.7 The power stage 17

Figure 2.8 Separately excited DC motor with current [4] 17

Figure 2.9 Vector-controlled induction motor [4] 18

Figure 2.10 Dq-winding equivalent circuit [5] 19

Figure 2.11 Dq-winding equivalent circuit in the synchronous reference frame while rotor flux is oriented with the d-axis [5] 19

Figure 2.12 Flowchart of the software 21

Figure 2.13 Stator d-axis current (blue) vs rotor flux (red) in the first 400ms . 22

Figure 2.14 Clarke transformation projects stator current vector to $\alpha\beta$ axis . 23

Figure 2.15 Abc reference frame to stationary reference frame transformation 24

Figure 2.16 Park Transform, d-axis is superposed with the rotor flux axis . . 26

Figure 2.17 Stationary reference frame to synchronous reference frame transformation 26

Figure 2.18 Conventional and series PI controller and their frequency responses [6] 27

Figure 2.19 Series PI controller in the synchronous reference frame 28

Figure 2.20 Discrete integrator implemented with bacward Euler method . . 29

Figure 2.21 Dynamic integrator clamping 30

Figure 2.22 Block diagram of axis decoupling 31

Figure 2.23 Block diagram of rotor flux observer 34

Figure 2.24 First-order IIR high-pass filter implementation 35

Figure 2.25 Schematic diagram of IPM with resistive braking IGBT 36

Figure 2.26 Two-level voltage source inverter topology 37

Figure 2.27 Possible switching states of VSI 38
Figure 2.28 Calculation of on times 39
Figure 2.29 Switching sequence in sector 1 40
Figure 2.30 The block diagram of the DC link compensation 41
Figure 2.31 Block diagram of synchronous speed calculation 42
Figure 2.32 Block diagram of slip speed and rotor speed sensing 43
Figure 2.33 Block diagram of the speed loop PI controller [7] 44
Figure 2.34 Asymmetric and symmetric dead-time insertion 45
Figure 2.35 Motor speed response under 3Nm load torque 46
Figure 2.36 I_{qsref} vs I_{qs} from Kayhan's thesis [3] 47
Figure 2.37 I_{qsref} vs I_{qs} for K_a=20 and K_b=400 47
Figure 2.38 I_{qsref} vs I_{qs} for K_a=20 and K_b=500 48

Figure 3.1 Block diagram of open-loop estimator 52
Figure 3.2 Block diagram of closed-loop estimator 53
Figure 3.3 Kalman filter block diagram 54
Figure 3.4 Probability density of the Kalman filter 55
Figure 3.5 Recursive algorithm of the Kalman filter 56
Figure 3.6 Taylor expansion of a nonlinear function 59
Figure 3.7 Recursive algorithm of the innovation based extended Kalman filter 63

Figure 4.1 IAEKF algorithm's subsystem in Simulink 66
Figure 4.2 IAEKF algorithm's implementation in Simulink 67

Figure 4.3 Convergence of the estimator when the initial condition for $\hat{R}_s$ $0\,\Omega$ 70

Figure 4.4 Convergence of the estimator when the initial condition for $\hat{R}_s$ is 4.45 Ω 70

Figure 4.5 Standstill simulation result for test motor without any series resistance 71

Figure 4.6 Standstill simulation result for test motor with 1 Ohm series resistance 71

Figure 4.7 Variables used to estimate $\mathbf{R_s}$ at 3000 RPM without 1 Ohm series resistance 72

Figure 4.8 Variables used to estimate $\mathbf{R_s}$ at 3000 RPM with 1 Ohm series resistance 73

Figure 5.1 Block diagram of the experimental setup conducted with an Inverter 77

Figure 5.2 Measurements done with PWM inverter at 50 Hz for different N and $R_k[A^2]$ values 80

Figure 5.3 Measurements done with PWM inverter at 50 Hz for different N and $R_k[A^2]$ values, 1 Ω series resistance added 80

Figure 5.4 R_s for different V_{rms} and different frequencies 81

Figure 5.5 Convergence of the estimator when the initial condition of $\hat{R}_s$ is $4.45\,\Omega$ 82

Figure 5.6 Convergence of the estimator when the initial condition of $\hat{R}_s$ is $0\,\Omega$ 82

Figure 5.7 Variables used to estimate $\mathbf{R_s}$ at 50 Hz without 1 Ω series resistance 83

Figure 5.8 Variables used to estimate $\mathbf{R_s}$ at 50 Hz with 1 Ω series resistance 84

Figure 5.9 Variables used to estimate $\mathbf{R_s}$ at 10 Hz without 1 Ω series resistance . . . 85

Figure 5.10 Variables used to estimate $\mathbf{R_s}$ at 10 Hz with 1 Ω series resistance 85

Figure 5.11 Variables used to estimate $\mathbf{R_s}$ at 25 Hz without 1 Ω series resistance . . . 86

Figure 5.12 Variables used to estimate $\mathbf{R_s}$ at 25 Hz with 1 Ω series resistance 87

Figure 5.13 Variables used to estimate $\mathbf{R_s}$ at 50 Hz without 1 Ω series resistance . . . 88

Figure 5.14 Variables used to estimate $\mathbf{R_s}$ at 50 Hz with 1 Ω series resistance 88

Figure 5.15 Variables used to estimate $\mathbf{R_s}$ at 75 Hz without 1 Ω series resistance . . . 89

Figure 5.16 Variables used to estimate $\mathbf{R_s}$ at 75 Hz with 1 Ω series resistance 90

Figure 5.17 Variables used to estimate $\mathbf{R_s}$ at 100 Hz without 1 Ω series resistance . . . 91

Figure 5.18 Variables used to estimate $\mathbf{R_s}$ at 100 Hz with 1 Ω series resistance . . . 91

Figure 5.19 Estimation of $\alpha\beta$ components of stator current $i_{s\alpha}$, $i_{s\beta}$ and stator flux $\varphi_{s\alpha}$, $\varphi_{s\beta}$ . . . 92

Figure 6.1 Variables used to estimate $\mathbf{R_s}$ at 600 RPM without 1 Ω series resistance . . . 95

Figure 6.2 Variables used to estimate $\mathbf{R_s}$ at 600 RPM with 1 Ω series resistance . . . 95

Figure 6.3 Variables used to estimate $\mathbf{R_s}$ at 1500 RPM without 1 Ω series resistance . . . 96

Figure 6.4 Variables used to estimate $\mathbf{R_s}$ at 1500 RPM with 1 Ω series resistance 96

Figure 6.5 Variables used to estimate $\mathbf{R_s}$ at 3000 RPM without 1 Ω series resistance 97

Figure 6.6 Variables used to estimate $\mathbf{R_s}$ at 3000 RPM with 1 Ω series resistance 98

Figure 6.7 Stator resistance estimation under 3 Nm load torque 100

Figure 6.8 Experimental setup 100

Figure 6.9 DS1104 ControlDesk software 101

Figure B.1 Clarke transforms implemented in Simulink 113

Figure B.2 Inverse Park transform implemented in Simulink 113

Figure B.3 Series PI controller implemented in Simulink 114

Figure B.4 The implementation of the DC link compensation in Simulink . . 114

Figure B.5 Rotor flux observer 114

Figure B.6 Synchronous speed calculation and IIR filter 115

Figure B.7 Slip speed and rotor speed calculation 115

Figure B.8 IGBT inverter in Simulink 116

Figure B.9 Motor model 116

Figure B.10 Torque calculation in Simulink 116

Figure B.11 Simulation file implemented in Simulink 117

Figure B.12 Simulation file modified and embedded in the DSPACE DS1104 118

LIST OF ABBREVIATIONS

AC	Alternative Current
DC	Direct Current
VSI	Voltage Source Inverter
VFD	Variable Frequency Drive
PWM	Pulse Width Modulation
SPWM	Sinusoidal Pulse Width Modulation
SVPWM	Space Vector Pulse Width Modulation
DSP	Digital Signal Processor
ADC	Analog to Digital Converter
EMF	Electro-Motive Force
IIR	Infinite Impulse Response
Hz	Hertz
PI	Proportional-Integral
IGBT	Insulated Gate Bipolar Transistor
IPM	Intelligent Power Module
RPM	Revolution Per Minute
EKF	Extended Kalman Filter
IAEKF	Innovation Based Adaptive Extended Kalman Filter
MOSFET	Metal Oxide Semiconductor Field Effect Transistor
MRAS	Model Reference Adaptive System
RFO	Rotor Flux Oriented Control
RTD	Resistance Temperature Detector

CHAPTER 1

INTRODUCTION

1.1 Electric Motors and Induction Motors

Electric motors have been in human life since the first half of the 19th century; the primary usage of electric motors is to convert electrical energy to mechanical energy. Their significance has become undeniable over the last 150 years in human civilization. For a long time, the DC motors were the primary choice of electric motors since a voltage source easily adjusted their torque and speed. However, the brushes make the DC motors vulnerable to maintenance issues and eliminate their use in highly reliable systems such as aerospace applications. On the other hand, AC induction motors do not require any brushes; the field is built at the stator, which induces currents in the rotor bars. By the Lorentz force, this field and current build torque output on the shaft, without any electrical contact between the rotor and the stator. This property of the induction motor eliminates the requirement of permanent magnet use at the rotor, making it very suitable and cheap for industrial and consumer applications. The traditional way to control an AC induction motor was without speed control. The motor is supplied directly from an electrical input, usually a utility grid; this makes the motor spin at its rated speed. However, this may result in unwanted results and performance. Firstly, motors become prone to utility line unbalances, resulting in faulty conditions, especially in precise applications such as chemical processes or textile manufacturing. Secondly, most of the industrial applications require distinct torque-speed characteristics than the motor's rated values; if this adjustment is made by mechanical control such as throttling valves, the process becomes very inefficient in terms of energy conversion. Nevertheless, after the invention of the semiconductor switches, microchips, new topologies, and algorithms, the AC motor drives, also known as variable fre-

quency drives (VFD) became available. The first proposed method was scalar control (V/f), which was simply achieved by keeping constant stator flux and the same voltage over frequency. With a mature mathematical model of the induction motor established, vector control is introduced. An AC motor can be easily controlled as a DC motor employing vector control or field-oriented control. Compared to scalar control, vector control helps an induction motor to have much better performance. After the invention of the vector control, researchers started to focus on this topic and try to increase the performance of the vector-controlled drive.

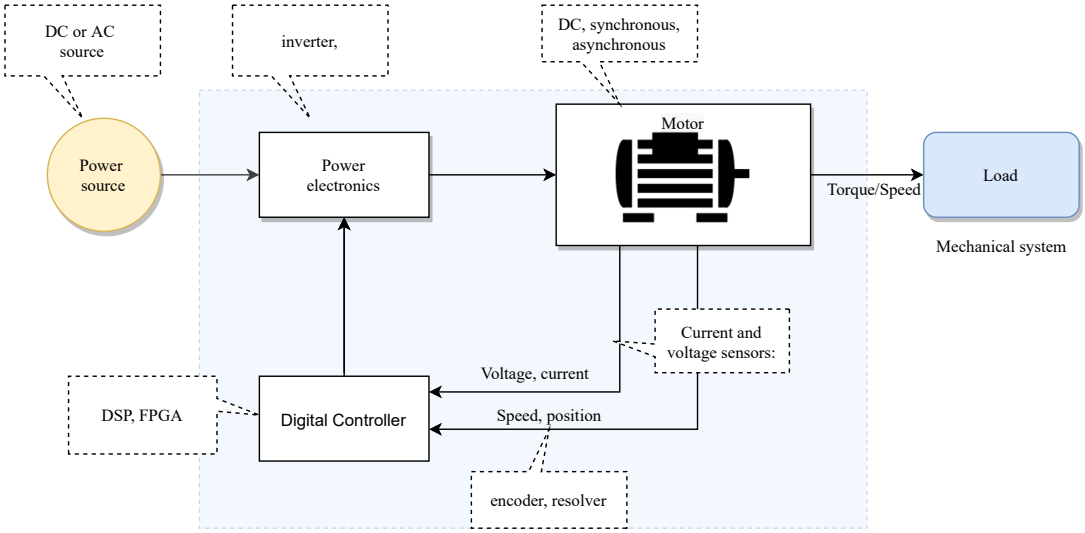


Figure 1.1: Block diagram of motor drives

The Voltage Source Inverter (VSI) is the most commonly used VFD in the industry and household electronics. Utility line voltage which is AC voltage is converted to DC voltage by a rectifier and filtered by a DC link capacitor. The inverter chops the DC link voltage so that the voltage and the output frequency are controlled; by this method, the motor's torque and speed can be controlled.

1.2 Vector Control

The vector control can be categorized into two main categories according to how the flux angle is obtained: direct and indirect vector control. The direct vector control was invented by Blashke [8] while indirect vector control was invented by Hasse [9]. In the direct vector control, the flux angle is either measured or estimated, while in

the indirect vector control the flux angle is obtained by integrating the sum of the slip speed and the rotor speed [4]. In this thesis, direct vector control is implemented, the rotor flux angle is estimated by an observer which is detailed in Chapter 2.

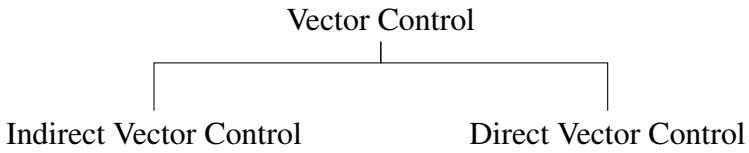


Figure 1.2: Vector control methods

The induction motor has three different fluxes available for vector control: rotor, stator, and air-gap flux. Among these reference fluxes, the rotor flux is chosen as reference most widely due to its simplicity, in this thesis, the rotor flux is chosen as reference flux. The vector control can be categorized into three main categories according to the reference flux vector [10]:

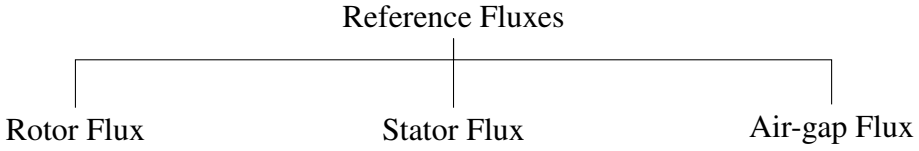


Figure 1.3: Reference fluxes available for vector control

In the applications where the cost is the primary concern, sensorless vector control is the main choice. Sensorless vector control refers to the lack of the position or speed sensor; on the other hand, voltage and current sensors exist in the drive system. In sensorless vector control, speed is estimated while in sensored vector control an encoder or resolver is mounted on the motor shaft in order to measure the rotor speed. The sum of rotor speed and slip speed is integrated to obtain the instantaneous flux angle, which is given in Equation 1.1.

$$\theta_e = \int_0^t \omega_e \, \mathrm{d}t = \int_0^t (\omega_{rotor} + \omega_{slip}) \, \mathrm{d}t \tag{1.1}$$

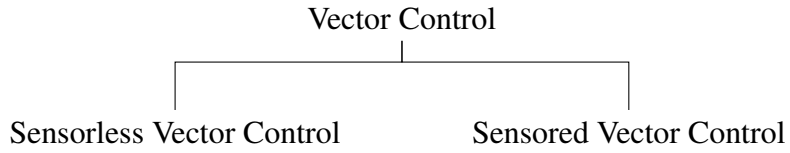


Figure 1.4: Vector control categorized according to position or speed sensor availability

1.3 Parameters and States That are Effective for High Dynamic Performance

The parameters and states that are effective for high dynamic performance are given as [3]:

- PI Controllers
- Integral Anti-windup Mechanism
- Axis Decoupling
- DC link Voltage
- Dead Time
- Switching Frequency
- Motor Parameter Accuracy

Kayhan [3] thoroughly investigated the dead time compensation, DC link compensation and DC link design procedure. The effects of the PI controller, the integral anti-windup mechanism, as well as axis decoupling are investigated in this thesis and a stator resistor estimation algorithm is added to accurately predict the rotor flux angle.

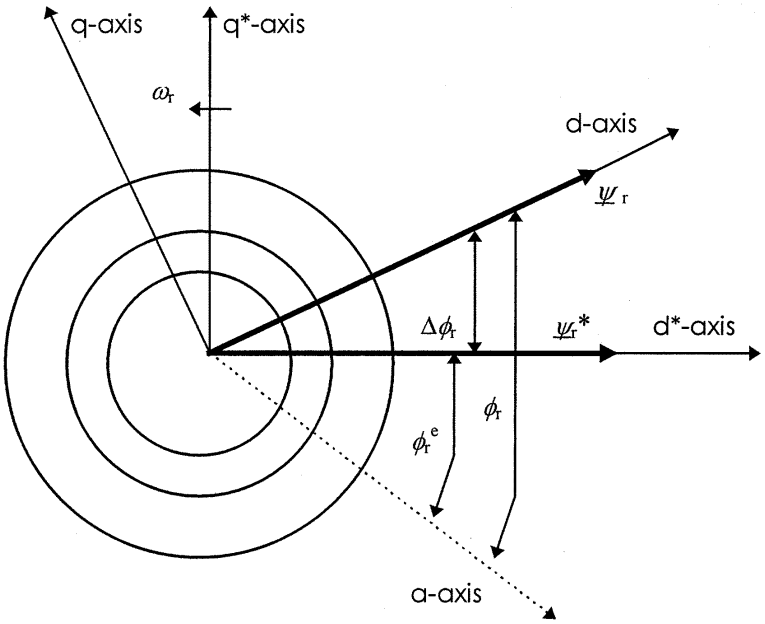


Figure 1.5: Actual vs commanded rotor flux in parameter mismatch situation [2]

1.4 Motor Parameter Estimation

When using vector control, it's vital that the DSP settings match the actual motor parameters. Furthermore, determining motor parameters using additional equipment may not always be feasible, mainly when the motor is large and outdoors. The no-load test and locked rotor test are traditional test procedures that are based on assumptions, resulting in loss of accuracy. The switching and harmonic effects are not taken into account in these tests. Another prevalent practice is that commercial motor drivers and motors in a system are usually made by different companies, in which case the motor'a parameters naturally do not pre-exist in the driver's memory. In such situations, to achieve high and satisfactory performance from the motor drive system, the parameters of the motor, which are crucial for vector control, have to be determined by the motor driver before the initialization of the motor [2]. The procedure of determining the initial motor parameters is known as self-commissioning for vector control.

The value of the stator resistance is crucial information when the direct vector control

algorithm is implemented. Equation 1.2 calculates the stator flux and the BACK-EMF voltage "$V_{s\alpha} - R_s i_{s\alpha}$" which can not be directly measured by any sensor. As seen in Equation 1.2, stator resistance value R_s has to be estimated accurately. If there is a mismatch between the motor and driver parameters, the actual and commanded rotor flux will differ as shown in Figure 1.5. The rotor flux estimation is studied in Chapter 2.4.11.

$$\lambda_{s\alpha} = \int (V_{s\alpha} - R_s i_{s\alpha})\, dt = \int e_\alpha dt \tag{1.2a}$$

$$\lambda_{s\beta} = \int (V_{s\beta} - R_s i_{s\beta})\, dt = \int e_\beta dt \tag{1.2b}$$

The parameter estimation techniques are divided into two categories:

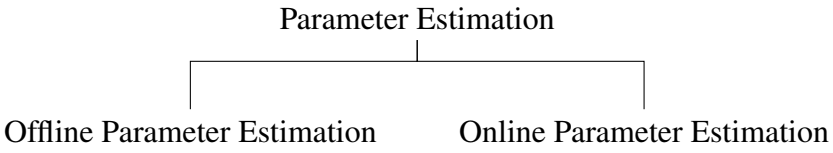


Figure 1.6: Parameter estimation

Offline parameter estimation is conducted when the motor is out of service. No-load test and blocked rotor test of induction motors are offline parameter identification tests, which are conducted with extra test equipment [11]. However these tests are not always practical since it requires extra test equipment. The cost of the extra equipment's forced the researchers to find new type of offline tests which can be done by motor connected to an inverter. Firstly, inverter generates specific signals and the output measurements are evaluated by an special algorithm so that there is no extra test equipment requirement. This second approach is more practical and most of the researchers are focused on this solution. Some offline parameter estimation techniques require to lock the rotor while others do not [12], [13], [14], [15], [16]. In online parameter estimation, the motor runs at rated speed and torque for a very long time so that the rotor and the stator windings become relatively hot with respect to starting condition, thus the estimation of the the stator resistance and rotor time constant becomes crucial for field-oriented control purposes during the operation.

Most of the researches focused on estimating stator resistance and rotor time constant [17], [18], [19].

In the literature researchers usually prefer different approaches for parameter estimation. These methods can be categorized as follows: observer-based estimation such as Kalman Filter and Extended Kalman Filter, MRAS (Model Reference Adaptive System) based estimators and artificial intelligence based estimators.

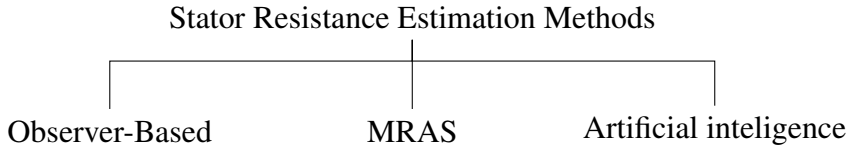


Figure 1.7: Stator resistor estimation techniques

Some researchers focused on Luenberger observer for stator resistance estimation. Reference [20] and [21] proposed a Luenberger observer for both rotor speed and stator resistance estimation.

Reference [22] proposed sliding mode observer based estimator for rotor speed, rotor resistance and stator resistance. The proposed method aims to solve accuracy problem of the motor parameters at low speed.

Model reference adaptive system (MRAS) is also studied by the researchers for the stator resistance estimation. MRAS consists of reference model and adaptive model, the difference of the reference model and adaptive model is defined as error, this error guides the adaptive model in a such a way that the adaptive model coincides the reference model. The designed system uses MRAS to estimate the rotor speed and the stator resistance in [23], the difference between the measured and observed current is used as an input to the adaptive model.

Reference [24] suggested a novel online identification approach for estimating both stator resistance R_s and rotor resistance R_r. The flux command is injected with a small magnitude and low frequency AC component in this approach. The steady state power flow between the stator and the rotor across the air gap led their R_s identification algorithm. This approach can only be used in steady-state situations. Furthermore, at low speeds, the injected low frequency signal may cause issues.

A further method for estimating the stator resistance is by using artificial intelligence techniques. Artificial intelligence approach for stator resistance estimation can be categorised into two groups: using artificial neural network and using fuzzy logic. In the neural network approach proposed by [25], on the other hand, the difference between measured and estimated stator currents is back-propagated to adjust the neural network weights to ultimately estimate the stator resistance. Reference [26] estimates the stator resistance online by fuzzy logic, taking the stator winding temperature, magnitude and frequency into account.

Compared to other techniques, EKF-based estimators account for the skin and proximity effect on the winding resistance, resulting in a more realistic stator resistance estimation. Due to the apparent superiority of EKF, there are studies in the literature which use various EKF-based estimators such as conventional EKF [27], unscented EKF [28] , fuzzy logic-based EKF [29] , augmented EKF [30], cascade EKF [31], adaptive fading-based EKF [32] , strong tracking EKF [33], symmetric strong tracking EKF and interfacing multiple-model EKF to perform stator resistance estimation.

Despite their improved estimating capabilities, these studies have considerable difficulty establishing the system and measurement noise covariance matrices (Q and R). Because of this setback, adaptive fading-based, strong tracking-based, symmetric-strong tracking-based, and multiple model-based adaptive approaches are more difficult to design than the IAEKF Innovation Based Adaptive Extended Kalman Filter (IAEKF), which does not require determining the Q matrix's initial constant values [34], [35].

In this thesis, sensorless vector control is performed using the induction motor's voltage model and it is necessary to provide the vector control algorithm with the correct stator resistor values to estimate rotor flux orientation. Furthermore, stator resistance values increase as the motor's temperature rises, making it difficult to accurately estimate the flux orientation of the rotor. The IAEKF algorithm is implemented for stator resistance estimation. The IAEKF algorithm is verified in simulation environment then tested in the real-time experimental setup.

1.5 The Outline of the Thesis

This thesis consists of seven chapters.

Chapter 1 conducts a literature review and outlines the parameters that determine the performance of a vector controlled drive.

Chapter 2 explains the hardware and the software of the vector control algorithm.

Chapter 3 explains the Kalman filter and innovation based adaptive extended Kalman filter (IAEKF).

Chapter 4 gives details about implementation of the innovation based adaptive extended Kalman filter (IAEKF) in Simulink.

Chapter 5 shows the experiment results of estimation of stator resistance at standstill.

Chapter 6 shows the experiment results of estimation of stator resistance at running condition.

Chapter 7 concludes the thesis with a summary of the work done herein, and discusses potential future work.

CHAPTER 2

THE HARDWARE AND THE SIMULATION ENVIRONMENT FOR VECTOR CONTROL

2.1 Introduction

To drive a three-phase AC electric motor drive, two-level voltage source inverter (VSI) is the primary choice since the magnitude and the frequency of the output voltage can be easily adjusted. The hardware utilized in this thesis is a custom-designed two-level VSI which has Insulated Gate Bipolar Transistor (IGBT) as a semiconductor switch. In this thesis, the hardware will not be explained in detail, however, the details can be found in the thesis work of Kayhan [3]. This hardware was designed to drive an AC motor whether it is an induction or permanent magnet motor both in scalar and vector control mode.

2.2 Brief Explanation of the Hardware

In this part of the thesis, some improvements on hardware, tests, and simulations of the hardware will be given to clarify the custom-designed hardware. The block diagram of the hardware is given in Figure 2.1.

The hardware can be briefly described as follows:

- The Simulink file designed to drive the motor is compiled and deployed into DSPACE DS1104 controller board.
- A computer with DSPACE ControlDesk software is used to communicate and control the DS1104 controller board.

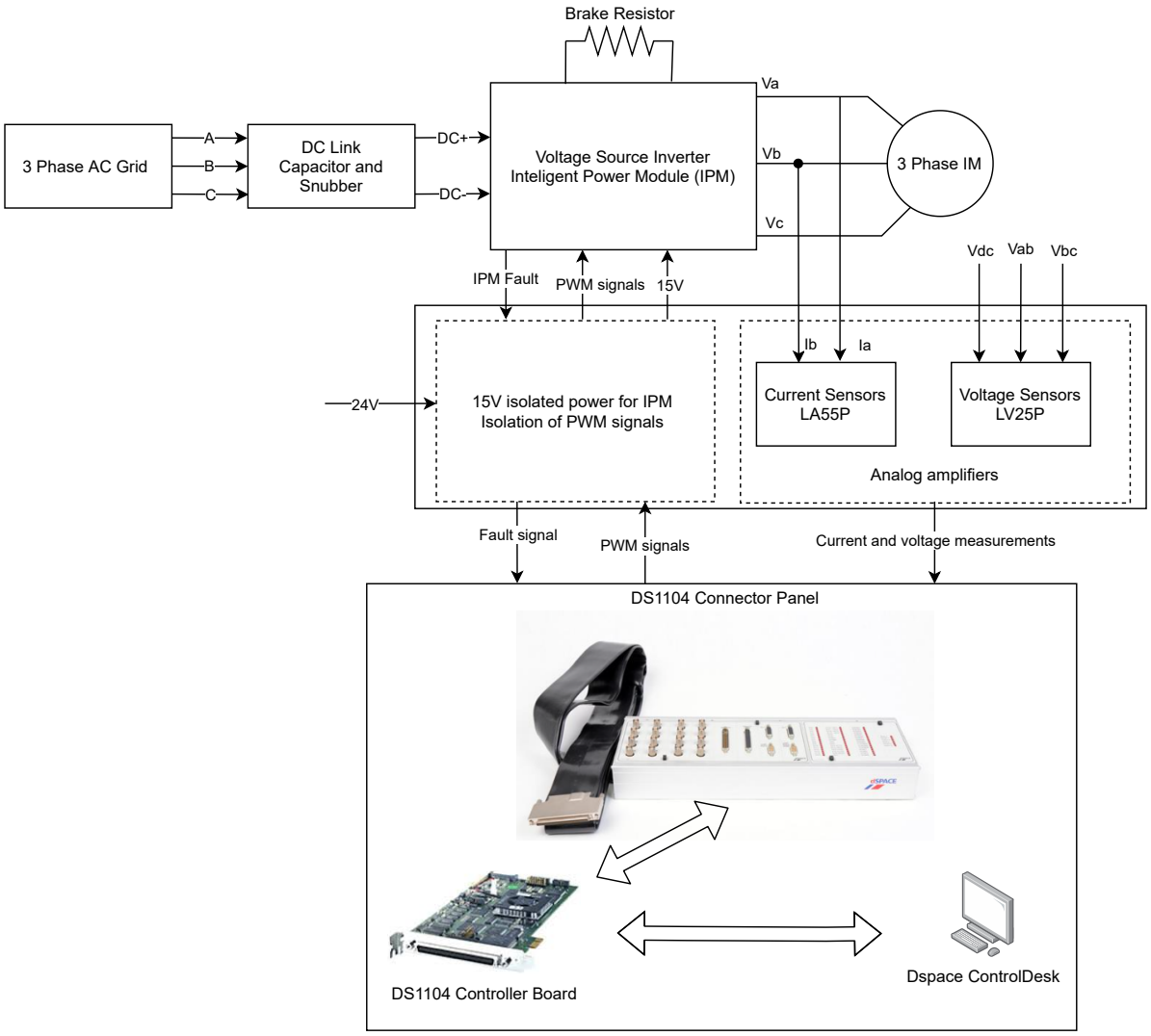


Figure 2.1: Block diagram of the hardware used in this thesis

- DS1104 connector panel has D-Sub and BNC connectors to connect the custom-designed hardware to the DS1104 controller board.

- The custom-designed PCB measures the currents and voltages by LA55P and LV25P and amplifies by Sallen-Key low-pass filter topology. These signals are fed to the BNC of the DS1104 controller board.

- PWM signals calculated at DS1104 are first buffered, then isolated by opto-couplers and send to the IPM which is Mitshubishi Electric's 1200V and 50A rated PM50RG1B120 model IPM. The IPM in this thesis is the new version of the IPM utilized in Kayhan's thesis [3]. The fault signal is also passed from an

optocoupler and send to the DS1104 controller board.

- High side IGBTs of the IPM are fed by a separate 15V isolated voltage source, on the other hand, low side IGBTs are fed by the same 15V isolated voltage source.

- The system is fed by a 380V AC grid, this voltage is rectified and converted to 537V DC and this voltage feeds the IPM. The IPM has 7 IGBTs, six of these IGBTs are utilized to form two-level voltage source inverter, the remaining IGBT is the braking IGBT.

2.2.1 Current and Voltage Sense Amplifier Circuits

In this thesis, some of the hardware blocks have been verified such as current and voltage measurement blocks. The design details of these circuits can be found in Kayhan's thesis [3]. The current sense amplifier circuit and the voltage sense amplifier circuit are similar. The current output of the sensor is converted to the voltage across a resistor. The output current of the sensor is converted to voltage by a resistor. This voltage drop is then buffered and filtered, then sent to the DS1104's analog to digital converter's (ADC). The current and the voltage sense amplifier circuits are identical, thus to explain the amplifier circuit's characteristics, current sense amplifier's circuit will be sufficient. The schematic diagram of the current sense amplifier is given in Figure 2.2.

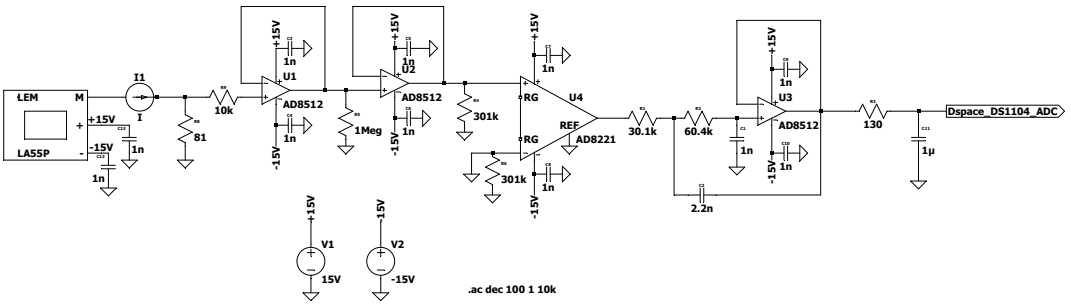


Figure 2.2: Current sense amplifier circuit's schematic diagram

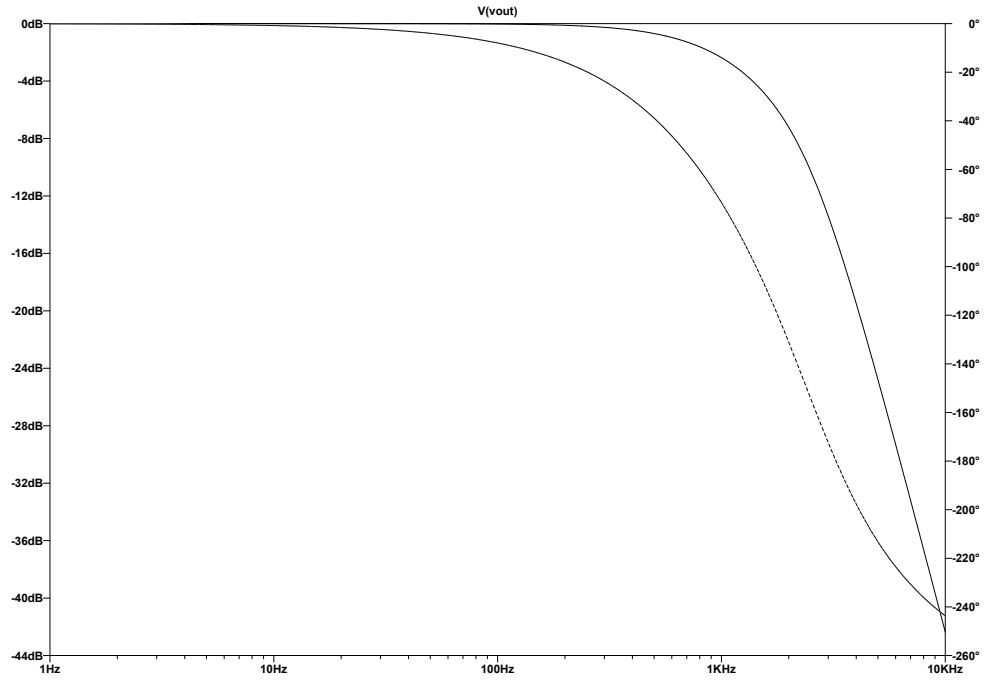


Figure 2.3: Frequency response of the current sense amplifier schematic diagram given in Figure 2.2

The circuit in Figure 2.2 is simulated in the LTspice ® which is a free and powerful circuit simulator that has an extensive component library. After the simulations, experiments on our custom hardware is conducted to find the frequency response of the filter circuits. In this experiments Keysight's DSOX1102G model oscilloscope which has frequency response analysis capability is utilized. Test setup shown in Figure 2.4 is built to measure the frequency response of the circuit. The signal generator of the DSOX1102G oscilloscope injects sweeping sinusoidal signals starting from 10Hz to 10 kHz with 1V peak to peak amplitude. The signal generator's output is connected to the filter circuit's input and at the same time both input and output of the filter circuits are probed simultaneously to determine the transfer function of the filter circuit.

The voltage and current amplifier circuits are critical components of the motor drive because if these measurements are not performed correctly, the IAEKF algorithm, which estimates stator resistance, will not produce accurate results. Furthermore, the actual and commanded rotor fluxes will mismatch, causing the motor to fail to produce the commanded torque.

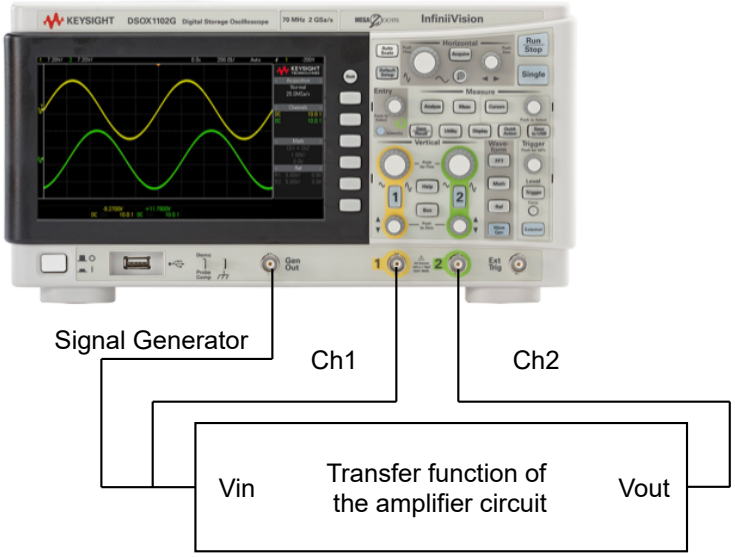


Figure 2.4: Experimental setup of frequency response analyze tests

The transfer function of the current sensor filter circuit is given in Figure 2.5. The cut-off frequency is found to be 5 kHz. Both current and voltage sensor filter circuits are tested and it is founded that all of these amplifier circuits have overlapping frequency response characteristics. The most important property of the sensor filter hardware is to have identical amplitude attenuation and identical phase delay for all of the sense amplifier circuits at a given frequency.

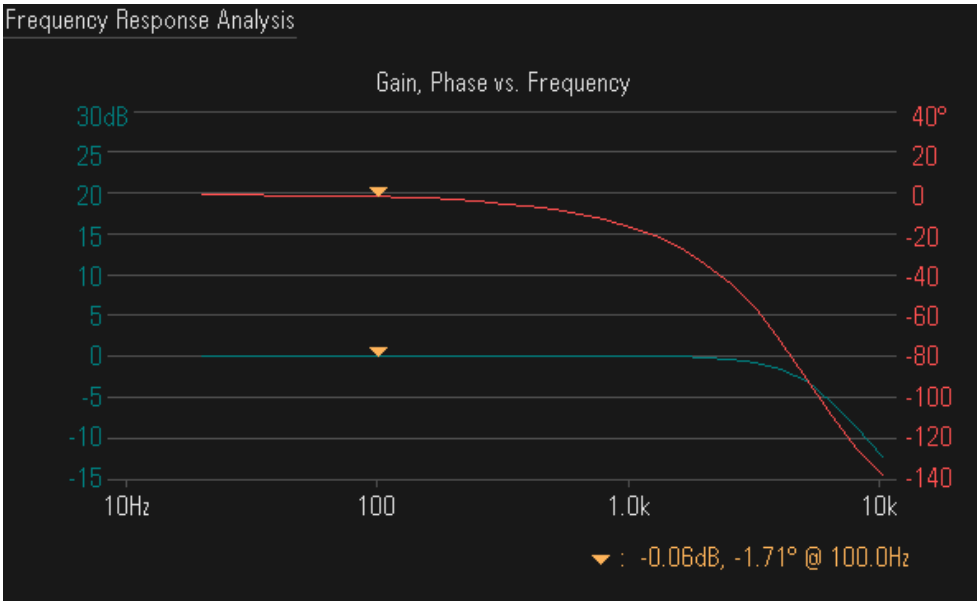


Figure 2.5: Transfer function of the current sensor filter circuit; blue: gain, red: phase

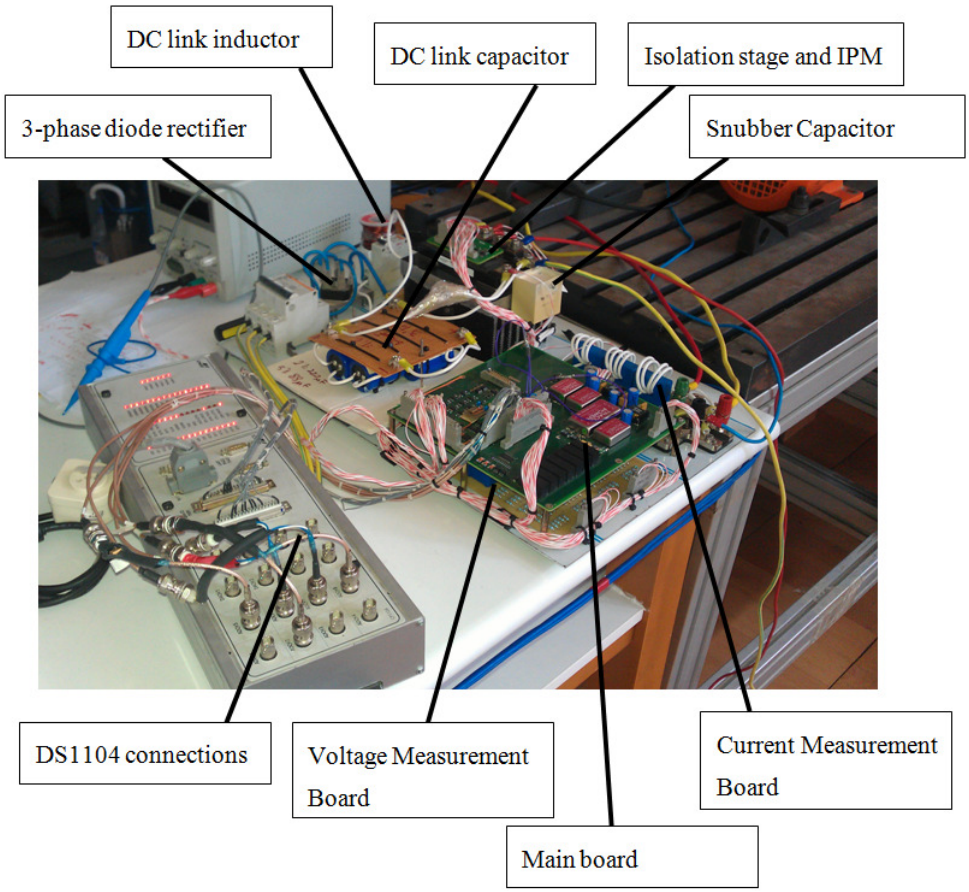


Figure 2.6: Motor drive hardware [3]

The hardware of the motor driver is given in Figure 2.6. For further details of the hardware, schematics and design details reader can refer to [3].

2.2.2 Power Stage and Motor

In the simulation environment, we have focused on modeling the power stage as accurately as possible, in Figure 2.7 three-phase AC source, diode rectifier, DC link, brake resistor and the IPM is modeled. The parameters used in the simulation of the diode and IPM are extracted from the data sheets.

The experimental motor's parameters were determined according to [11], same parameters are used in the simulation to get accurate simulation results. The parameters of the motor used in the simulations and in the experiments are given in Appendix A.

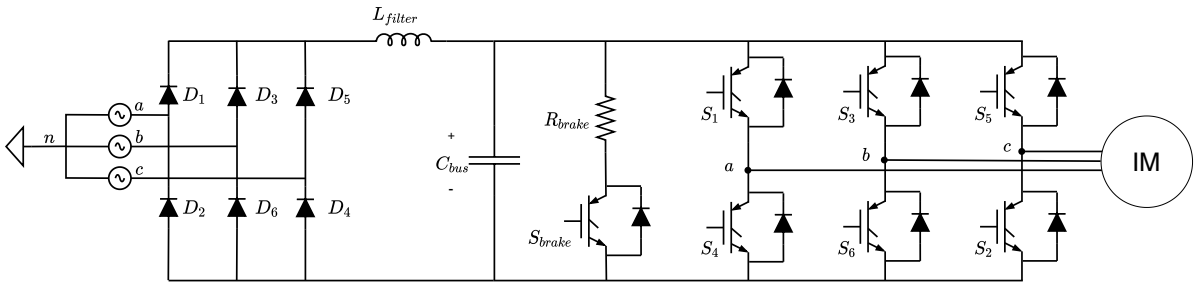


Figure 2.7: The power stage

2.3 Vector Control

The objective of vector control of an induction motor is to emulate the behavior of a separately excited DC motor. A separately excited DC motor's magnetic field and torque are directly and independently regulated by the field and armature currents. For a constant field current, the armature current uniquely determines the resulting electromagnetic torque. Furthermore, the field and the armature windings create two magnetic fields that are naturally perpendicular, giving rise to the most favorable condition for torque production. In Figure 2.8, it can be seen that the field flux and armature current are perpendicular to each other, the field current is adjusted to establish the flux, while the armature current is adjusted to set the torque. Both of these currents have no influence on each other since they are spatial in space. Equation 2.1 is the torque output of the DC motor.

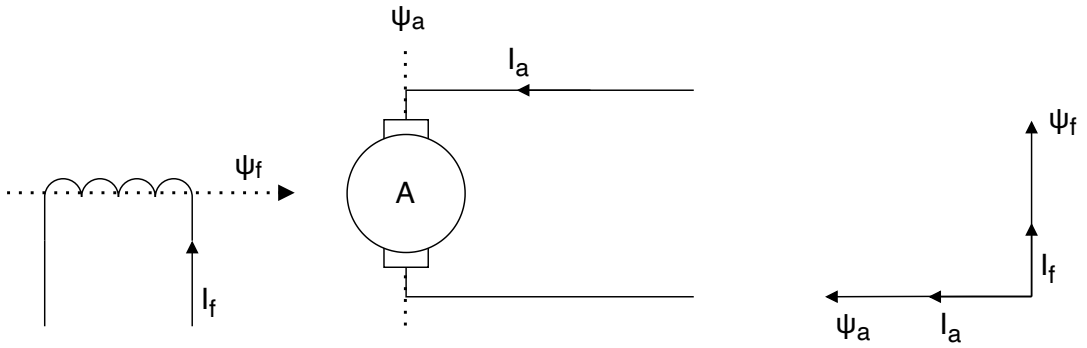


Figure 2.8: Separately excited DC motor with current [4]

$$T_e = K_t I_a I_f \tag{2.1}$$

Similarly, in an induction motor, the fundamental contributor for establishing a magnetic field and electromagnetic torque is the stator current. Vector control takes advantage of this absolute quantity in a synchronously rotating reference frame to obtain independent adjustment of the field flux and the torque. In the synchronously rotating reference frame, the AC quantities show up as DC quantities. Figure 2.9 shows that the induction motors can be similarly controlled as a DC motor when $\mathbf{I_{ds}}$ is aligned with the rotor flux $\lambda_{\mathbf{r}}$ and I_{qs} is perpendicular to $\lambda_{\mathbf{r}}$. In this condition, $\mathbf{I_{ds}}$ become flux establishing current, while the I_{qs} become the torque establishing current. Equation 2.2 shows the calculation of induction motor's electrical torque output.

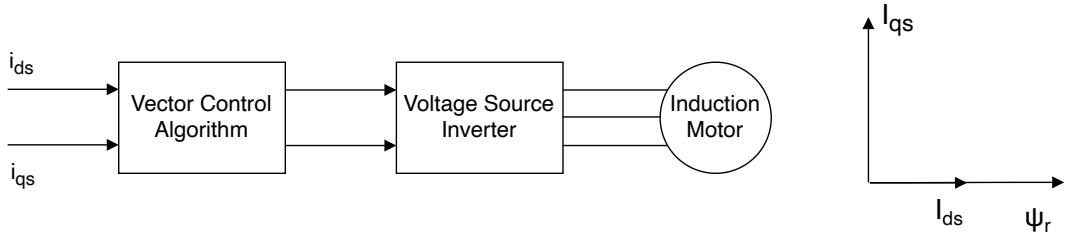


Figure 2.9: Vector-controlled induction motor [4]

$$T_e = K_t I_{ds} I_{qs} \tag{2.2}$$

2.3.1 Vector Control of Induction Motor

In this thesis rotor flux-oriented control (RFO) is implemented, thus d-axis is aligned with the rotor flux linkage vector. For its simplicity, RFO is by far the most commonly used vector control method. In this part, the RFO will be explained and the reason behind the transforms in the previous sections will be understood clearly. Figure 2.10 shows the induction motor's synchronous reference frame equivalent circuit.

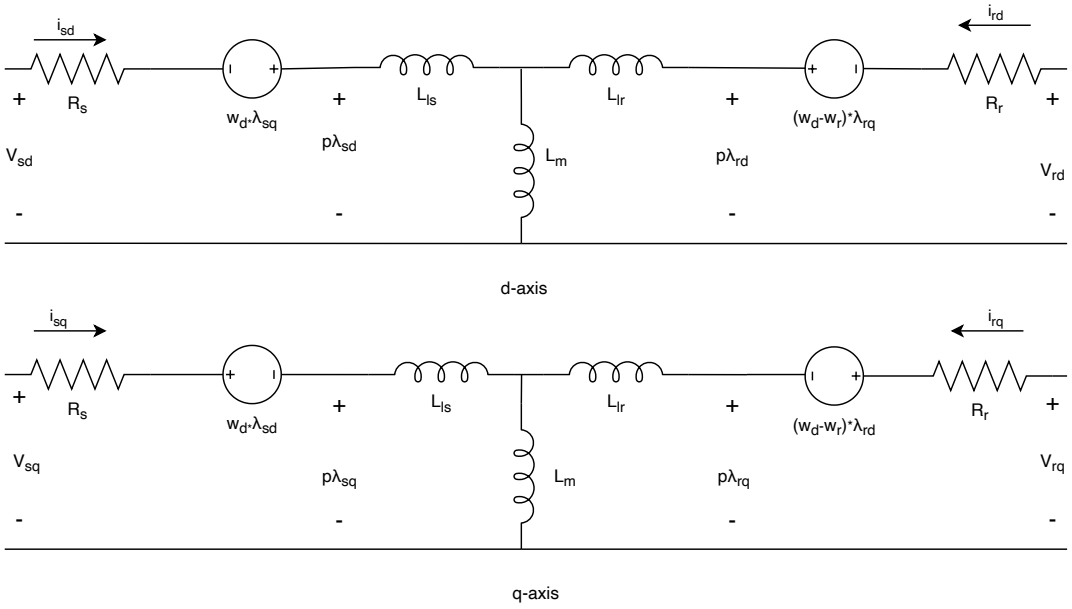


Figure 2.10: Dq-winding equivalent circuit [5]

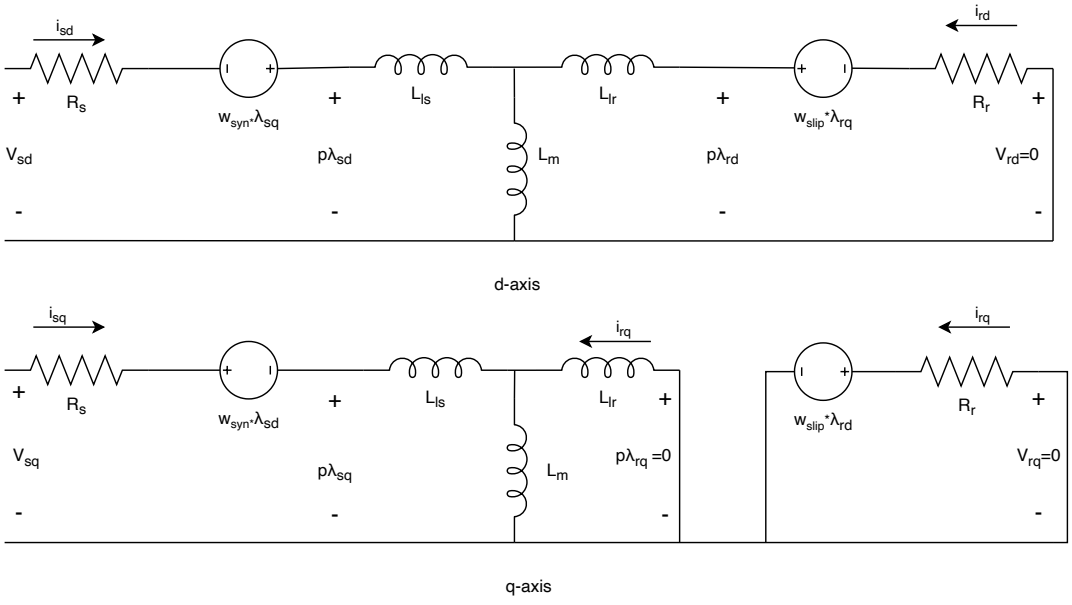


Figure 2.11: Dq-winding equivalent circuit in the synchronous reference frame while rotor flux is oriented with the d-axis [5]

2.4 Software Environment

In this thesis, the vector control algorithm is first implemented in Simulink, before embedding the real-time software to the DS1104 controller board, which is also programmed by the very same Simulink file. The DS1104 controller board is compatible with Matlab 2019b version. In this part of the thesis the vector control algorithm implemented on software environment will be explained. And the parameters that affect the performance of the vector-controlled drive will be investigated.

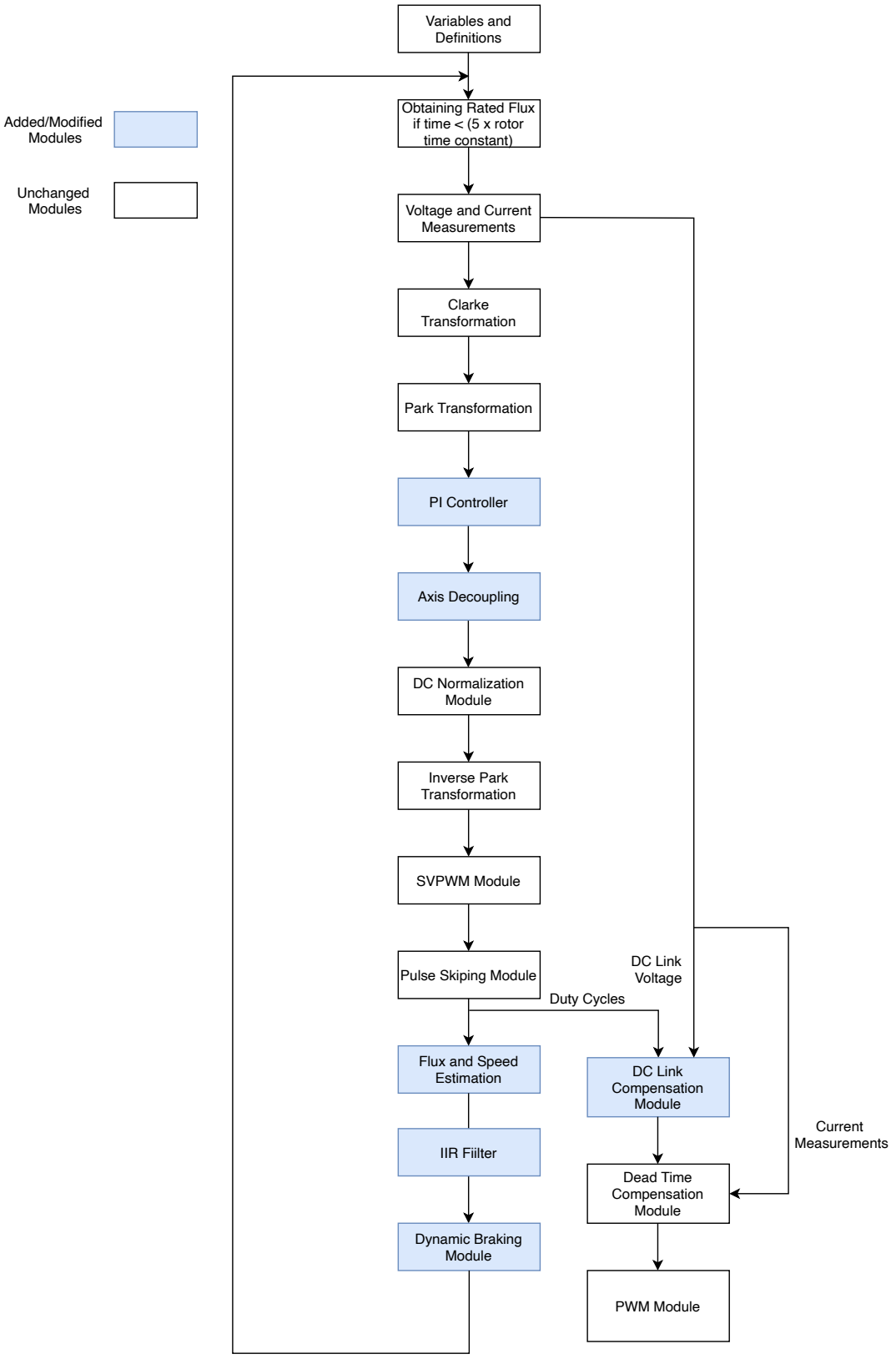


Figure 2.12: Flowchart of the software

2.4.1 Variable Definitions and Initialization Module

In this module, the motor parameters and variables used in the Simulink file are defined. All of the parameters are defined in the model workspace part of the Simulink file. The parameters of the experimental motor is given in Appendix A.

2.4.2 Obtaining the Rated Flux

In this module, the first 400ms of the time is dedicated to build the rated flux of the rotor. Stator d-axis current i_{sd} is applied at its rated value which is $2.4A$ DC current, while i_{sq} is applied as zero. In this time period, the d-axis is aligned with the a-axis. In Equation 2.3 the relation between the i_{ds} and λ_{rd} is given, this equation is a first-order differential equation. The rotor flux λ_{rd} reaches to its final value in $5\tau_r$ time, where $\tau_r = L_r/R_r$. After the steady-state is reached the relation between the rotor flux λ_{rd} and the i_{sd} is given in Equation 2.4. As an outcome, i_{sd} is generally referred to as flux-producing current. The calculation of i_{sdref} current is detailed in Appendix A.

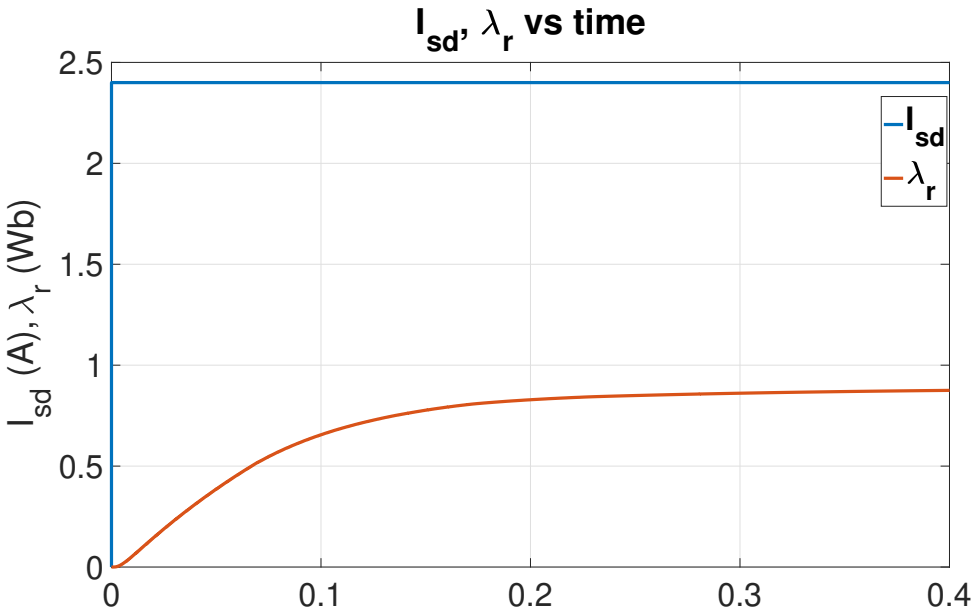


Figure 2.13: Stator d-axis current (blue) vs rotor flux (red) in the first 400ms

$$\lambda_{rd}(s) = \frac{L_m}{1 + s\tau_r} i_{sd}(s) \tag{2.3}$$

$$\lambda_{rd}(s) = L_m i_{sd}(s) \tag{2.4}$$

2.4.3 Voltage and Current Measurement

This module is different in the real-time and the simulation software, in the simulation the output of the voltage and sensors are not scaled, these are directly fed to the related blocks. On the other hand, in the real-time software, we have to take into consideration the sensor, amplifier and analog circuits of the DSP to set the sensor block's gains.

2.4.4 Clarke Transformation Module

This part of the software transformation is responsible for transforming the three-phase "abc" variables to "$\alpha\beta$" stationary reference frame. The i_{sd} vector can be represented by the combination of two phase currents instead of three-phase currents by transforming to $\alpha\beta$ reference frame with the Clarke transform. With the Clarke transform the number of equations are decreased by % 33.

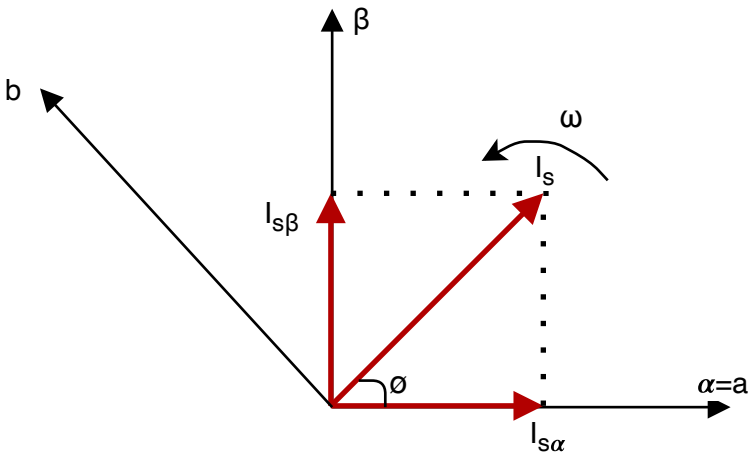


Figure 2.14: Clarke transformation projects stator current vector to $\alpha\beta$ axis

The current vector transformations are given as follows in Equation 2.5.

$$I_{s\alpha} = i_{sa} \tag{2.5a}$$

$$I_{s\beta} = \frac{1}{\sqrt{3}} i_{sa} + \frac{2}{\sqrt{3}} i_{sb} \tag{2.5b}$$

Since the measured voltages are phase to phase voltages, Equation 2.6 converts these measurements to phase voltages.

$$v_{sa} = \frac{(2v_{sab} + v_{sbc})}{3} \tag{2.6a}$$

$$v_{sb} = \frac{(v_{sbc} - v_{sab})}{3} \tag{2.6b}$$

Equation 2.7 is used to transform voltage vectors to stationary reference frame once the phase voltages have been obtained in Equation 2.6. The outputs of these blocks are used as input to the flux observer module and Park transform module.

$$V_{s\alpha} = v_{sa} \tag{2.7a}$$

$$V_{s\beta} = \frac{1}{\sqrt{3}} v_{sa} + \frac{2}{\sqrt{3}} v_{sb} \tag{2.7b}$$

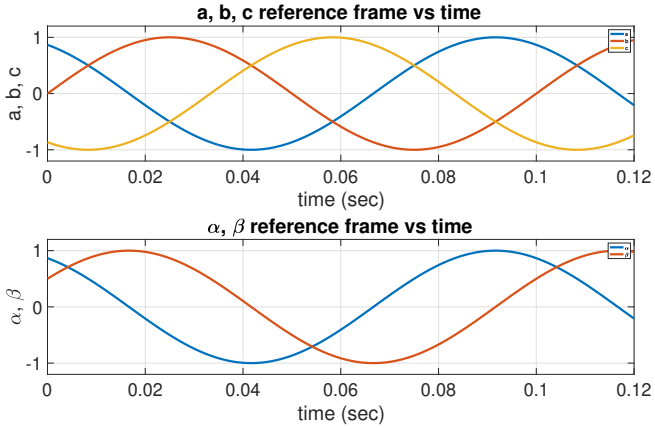


Figure 2.15: Abc reference frame to stationary reference frame transformation

2.4.5 Park Transformation Module

As the space vector rotates α and β variables vary in a sinusoidal pattern; the waveform of these variables can be seen in Figure 2.17. Stationary reference frame to synchronous reference frame transform which is called the Park transform can make the control of the motor simpler since the sinusoidal waveform does not exist anymore in the synchronous reference frame. The rotor flux is synced with the synchronous reference frame's d-axis and the space vector's reflection on the d and q axes becomes purely DC quantities. As a result, the vector control of an induction motor becomes as straightforward as control of a DC motor [10]. The Equation 2.8 implements Park transform for the current variables, while Equation 2.9 implements Park transform for the voltage variables. Figure 2.16 demonstrates the relation between the synchronous reference frame and the stationary reference frame.

$$I_{sd} = I_{s\alpha} \cos\theta + I_{s\beta} \sin\theta \tag{2.8a}$$

$$I_{sq} = -I_{s\alpha} \sin\theta + I_{s\beta} \cos\theta \tag{2.8b}$$

$$V_{sd} = V_{s\alpha} \cos\theta + V_{s\beta} \sin\theta \tag{2.9a}$$

$$V_{sq} = -V_{s\alpha} \sin\theta + V_{s\beta} \cos\theta \tag{2.9b}$$

Figure 2.17 demonstrates the stationary reference frame to synchronous reference frame transformation, as can be seen in the figure; in the synchronous reference the variables become DC quantities:

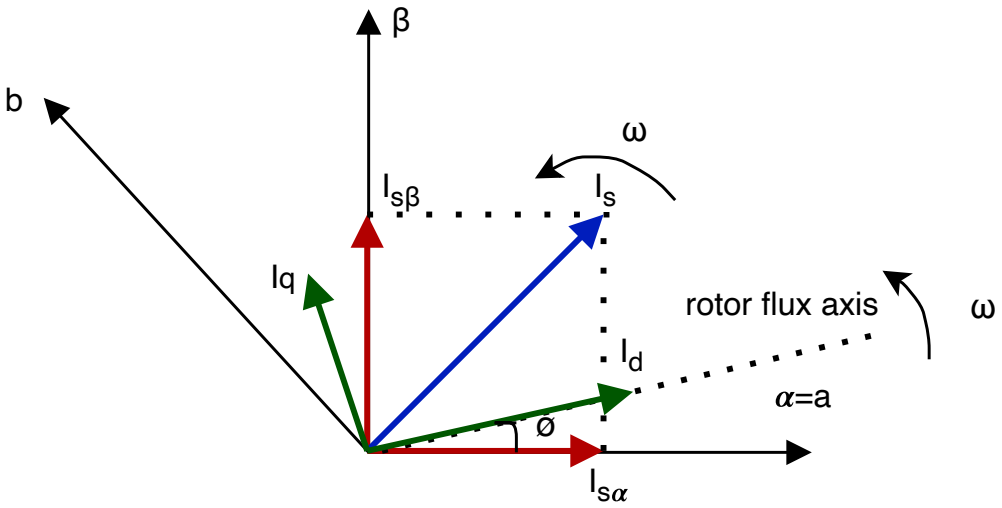


Figure 2.16: Park Transform, d-axis is superposed with the rotor flux axis

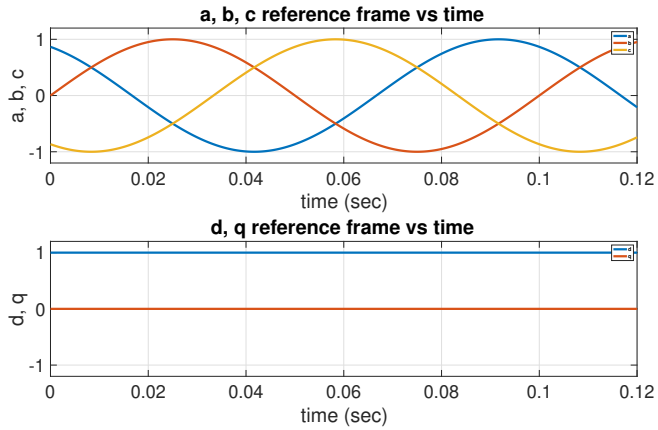


Figure 2.17: Stationary reference frame to synchronous reference frame transformation

2.4.6 PI Controller Module

The rotor flux is aligned with the d-axis and spins at synchronous speed in the rotor flux-oriented control; all variables are DC quantities in this frame of reference; thus the induction motor's current control becomes quite similar to the current control of a DC motor. The regulation of the i_{sd} and i_{sq} currents are achieved by the PI controller. In Equation 2.3 it can be seen that i_{sd} is responsible for the building the λ_{rd}. For up to rated speed, i_{sd} current is calculated to be $2.4A$. In Figure 2.16 it is seen that i_{sq} is

perpendicular to the rotor flux, q axis component of the synchronous reference frame is responsible for the torque production. The output of the PI controller module is V_{sdref} and V_{sqref} voltages. In Figure 2.18 conventional PI controller for d and q axes are given. However, in this thesis, we prefer to use series PI controller since it is much easier to tune.

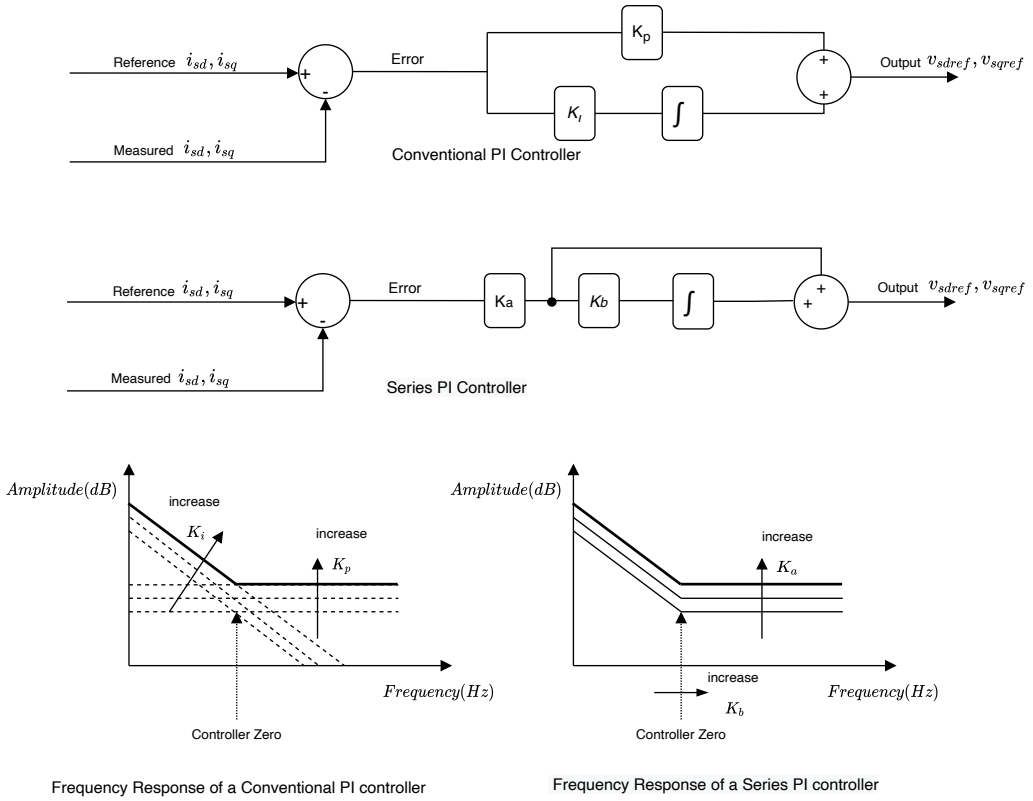


Figure 2.18: Conventional and series PI controller and their frequency responses [6]

The value that makes the numerator of the transfer function to be zero is defined as the zero of the transfer function, in Figure 2.18 the breaking point is the location of the zero of the controller. To adjust the zero location of the conventional PI controller, K_i, and K_p terms have to be modified at the same time, on the other hand for the adjustment of the zero location of the series PI controller, changing only the K_b value is adequate. The relation between K_i, K_p, and K_a, K_b is given in Equation 2.10.

$$K_a = K_p \tag{2.10a}$$

$$K_b = \frac{K_i}{K_p} \tag{2.10b}$$

Figure 2.19 shows the PI controller and the motor model in the synchronous reference

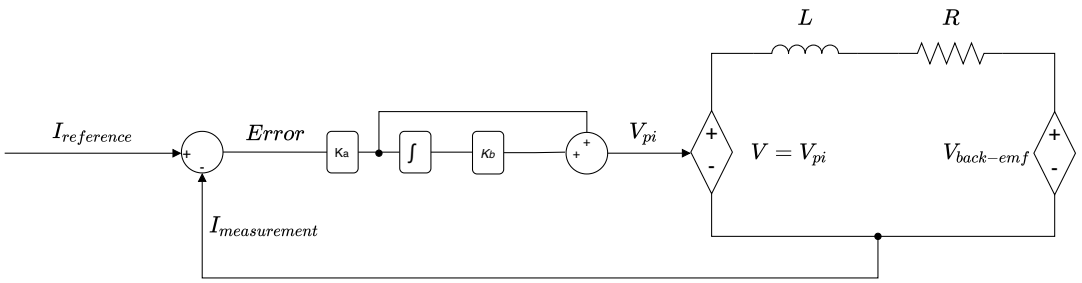


Figure 2.19: Series PI controller in the synchronous reference frame

frame. As it can be observed the induction motor is very similar to the DC motor model in the synchronous reference frame. As the location of the PI controller's zero cancels the pole of the plant, we are starting to deal with a first-order system. To have a pole-zero cancellation and a first-order system response, the PI controller's zero is chosen to cancel one of the poles of the RL circuit in Figure 2.19. The coefficients for pole-zero cancellation are given in Equation 2.11 [7]. The disadvantage of pole-zero cancellation method is that the resistance value utilized in Equation 2.11 can change with the temperature. This make the pole-zero cancellation method unreliable, the pole location may not exactly cancel the corresponding pole and the system response may become unstable. This issue can be solved by updating the resistance value in the DSP during the operation. In Section 2.5, experiments results for different PI coefficients are given.

$$K_a = L * \omega_{bandwidth} \tag{2.11a}$$

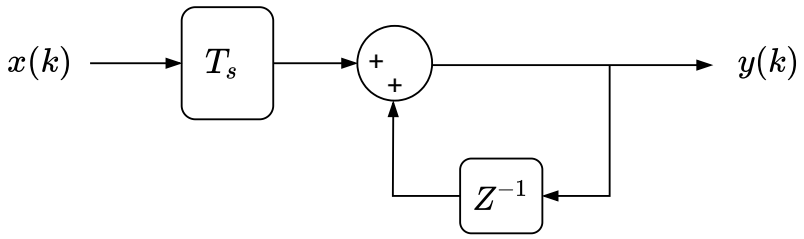
$$K_b = \frac{R}{L} \tag{2.11b}$$

To implement integral operation in a DSP the continuous integral has to be altered to discrete form by one of the methods shown in Table 2.1.

We have preferred the backward Euler method while implementing the digital integrator of the PI controller. Figure 2.20 shows the block diagram of the digital integrator.

Table 2.1: Discretization methods to implement the integration [1]

Discretization method	z -transform	Difference equation
Forward Euler	$H(z) = T_s \left(\frac{z^{-1}}{1-z^{-1}}\right)$	$y[n] = y[n-1] + T_s \cdot x[n-1]$
Backward Euler	$H(z) = T_s \left(\frac{1}{1-z^{-1}}\right)$	$y[n] = y[n-1] + T_s \cdot x[n]$
Trapezoidal	$H(z) = \frac{T_s}{2} \left(\frac{1+z^{-1}}{1-z^{-1}}\right)$	$y[n] = y[n-1] + \frac{T_s}{2} \cdot (x[n] + x[n-1])$



$$y(k) = x(k) * T_s + y(k-1)$$

Figure 2.20: Discrete integrator implemented with bacward Euler method

2.4.7 Anti-Windup Mechanism

Integral windup happens as a physical system's constraint when contrasted to ideal systems, because the perfect output is physically not possible. Every control system is analyzed with the assumption of linear models; however, in real life, there are some limitations to these systems. In our electric drive system, the limit of the voltage applied is affected by the motor's rated voltage, IGBT's breakdown voltage and the DC link voltage. Since these physical limits exist in our drive system, we have to limit the applied voltage to the motor to 311V. The calculation of this limitation can be found in Section 2.4.9. The saturation block breaks the feedback loop and the output voltage of the saturation block is limited to 311V no matter what the controller output value is. While the saturation block limits the output voltage to 311V, the PI controller's integrator continues to grow up since it continuously integrates the error. When the output voltage of the controller comes out of the saturation, this accumulated error cause a wild overshoot in the system. This event is known as integral windup. There are several integral anti-windup methods [36], the most popular integral anti-windup methods are integrator clamping, back-calculation and observer approach. The main idea in each integral anti-windup method is to limit the integrator value inside the sat-

uration limit values so that when a sign change in error occurs, the integrator responds to this change as fast as possible.

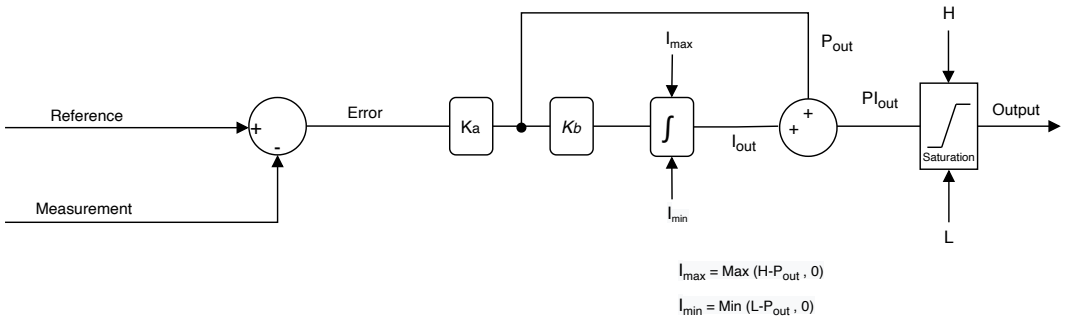


Figure 2.21: Dynamic integrator clamping

The integrator clamping set the limits to the maximum and minimum values of the saturation block so it guarantees that the windup event does not occur. In our drive system we preferred to use dynamic integrator clamping method, the most significant difference of the dynamic integrator clamping is that it subtracts the proportional controller's output value from the upper and lower limit of the saturation block, since the integrator cannot effect the system up to these levels. The dynamic integrator clamping causes less overshoot and has better dynamic response compared to the other methods [37]. The block diagram of dynamic integrator clamping is shown in Figure 2.21.

2.4.8 Axis Decoupling

The outputs of axis decoupling blocks are added to PI controller's output. These blocks decouple the d and q axes. The PI controller is implemented in the synchronous reference frame, as it can bee seen in Equation 2.22a and Equation 2.22b, the voltage expression of an induction motor in the d-q reference frame has cross-coupling terms, which depends on the motor speed and d-q axes fluxes of the motor. When making change in one axis rapidly, that change is reflected by these cross-coupling terms into to other axis. In the high frequencies and when the bandwidth of the controller is not large enough, the cross-coupling terms become problematic. To compensate the effect of these disturbances, feedforward control can be applied

to each axis along the feedback loop, to eliminate the cross-coupled voltage terms precisely. The topology represented in Figure 2.22 shows the calculation of the correction voltages to be applied to V_{sdref} and V_{sqref} voltages. These applied voltages to the V_{sdref} and V_{sqref} rejects out the cross-coupling terms [5], [37], [10].

$$\mathrm{i_d}\left(\mathrm{R_s} + \mathrm{DL_s}\sigma\right) = \mathrm{V_d} + \omega_e \mathrm{L_s}\sigma \mathrm{i_q} - \frac{\mathrm{L_m}}{\mathrm{L_r}} \frac{d\lambda_{\mathrm{rd}}}{dt} \tag{2.12a}$$

$$\mathrm{i_q}\left(\mathrm{R_s} + \mathrm{DL_s}\sigma\right) = \mathrm{V_q} - \omega_e \mathrm{L_s}\sigma \mathrm{i_d} - \omega_e \frac{\mathrm{L_m}}{\mathrm{L_r}} \lambda_{\mathrm{rd}} \tag{2.12b}$$

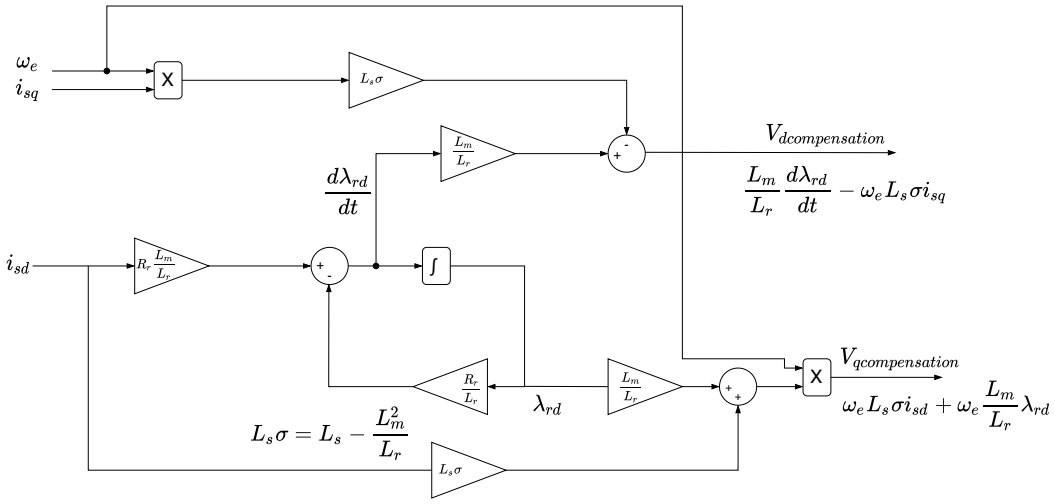


Figure 2.22: Block diagram of axis decoupling

2.4.9 DQ Voltage Limiter Module

This module limits the V_{sdref} and V_{sqref} voltages to 311V in the synchronous reference frame. If the inverter limit is not reached the normalization outputs the same input as in Equation 2.13, if the limit is reached normalization is done according to Equation 2.14. The 311V is the limit voltage for the inverter, this will be explained in Section 2.4.14. This algorithm is implemented with Matlab function and it is located after the PI controller.

- If $\sqrt{(V_{sdref})^2 + (V_{sqref})^2} \leq 311V$

$$V_{sdrefnorm} = V_{sdref} \tag{2.13a}$$

$$V_{sqrefnorm} = V_{sqref} \tag{2.13b}$$

- If $\sqrt{(V_{sdref})^2 + (V_{sqref})^2} > 311V$

$$V_{sdrefnorm} = \frac{V_{sdref}}{\sqrt{(V_{sdref})^2 + (V_{sqref})^2}} \times 311V \tag{2.14a}$$

$$V_{sqrefnorm} = \frac{V_{sqref}}{\sqrt{(V_{sdref})^2 + (V_{sqref})^2}} \times 311V \tag{2.14b}$$

2.4.10 Inverse Park Transformation Module

This module gets the inputs V_{sdref} and V_{sqref} voltages and transform these inputs to the stationary reference frame voltages $V_{sref\alpha}$ and $V_{sref\beta}$. Equation 2.15 shows the inverse Park transform equations.

$$V_{sref\alpha} = V_{sdref}\cos\theta - V_{sqref}\sin\theta \tag{2.15a}$$

$$V_{sref\beta} = V_{sdref}\sin\theta + V_{sqref}\cos\theta \tag{2.15b}$$

2.4.11 Rotor Flux Angle Estimation Module

The most important information for implementation of the vector control algorithm is the orientation of the rotor flux linkage at a specific time. The rotor flux orientation information is significant for Park and inverse Park transforms and these transforms are crucial for the vector control. To find the rotor flux angle, we need to calculate the rotor's flux linkage components in the stationary reference frame [10]. There are two methods to calculate the flux linkage components in the stationary reference frame. The first method uses hall effect sensors and sensing coils to directly measure the flux linkage. The second method uses motor models to estimate the magnitude and orientation of the flux linkage. In this thesis we utilized induction motor's voltage model to estimate the flux position. First of all the stator flux's stationary reference

frame components $\lambda_{s\alpha}$ and $\lambda_{s\beta}$ are obtained by the induction motor's voltage model. Then the stationary reference frame components of the rotor flux linkages are $\lambda_{r\alpha}$ and $\lambda_{r\beta}$ obtained and the rotor flux linkage orientation is calculated from these. The stator voltage equation is the combination of voltage drop at the stator resistance and the flux linkage's change rate, which is given in Equation 2.16.

$$V_{s\alpha} = R_s i_{s\alpha} + \frac{d\lambda_{s\alpha}}{dt} \tag{2.16a}$$

$$V_{s\beta} = R_s i_{s\beta} + \frac{d\lambda_{s\beta}}{dt} \tag{2.16b}$$

Integrating the flux linkage gives Equation 2.17, this equation estimates the back-EMF of the motor.

$$\lambda_{s\alpha} = \int (V_{s\alpha} - R_s i_{s\alpha})\, dt = \int e_\alpha dt \tag{2.17a}$$

$$\lambda_{s\beta} = \int (V_{s\beta} - R_s i_{s\beta})\, dt = \int e_\beta dt \tag{2.17b}$$

After the integration of the stationary reference frame components of the back-EMF voltage "$V_s - R_s i_s$" in Equation 2.17, there should be a high-pass filter due to DC offsets of the sensors, which can accumulate at the integral operation and result in false calculations. We have implemented a first-order high-pass IIR filter shown in Figure 2.24 to eliminate DC offsets which exists due to the current sensors and analog circuits. The first-order IIR filter is placed after the calculation of $\lambda_{r\alpha}$ and $\lambda_{r\beta}$, which is shown in Figure 2.23. The cut-off frequency of the filter is set equal to the rotor time constant. The total leakage factor is defined in Equation 2.18.

$$\sigma = 1 - \frac{L_m^2}{L_s L_r} \tag{2.18}$$

After the integration and the filtering with the high-pass IIR filter, the rotor flux linkages are calculated with Equation 2.19. The θ angle which is very crucial for the vector control is calculated by the arctan of the rotor flux linkage's stationary reference frame components $\lambda_{r\alpha}$ and $\lambda_{r\beta}$.

$$\lambda_{r\alpha} = \frac{L_r}{L_m}\left(\lambda_{s\alpha} - \sigma L_s i_{s\alpha}\right) \tag{2.19a}$$

$$\lambda_{r\beta} = \frac{L_r}{L_m}\left(\lambda_{s\beta} - \sigma L_s i_{s\beta}\right) \tag{2.19b}$$

$$\theta = \arctan(\frac{\lambda_{r\beta}}{\lambda_{r\alpha}}) \tag{2.19c}$$

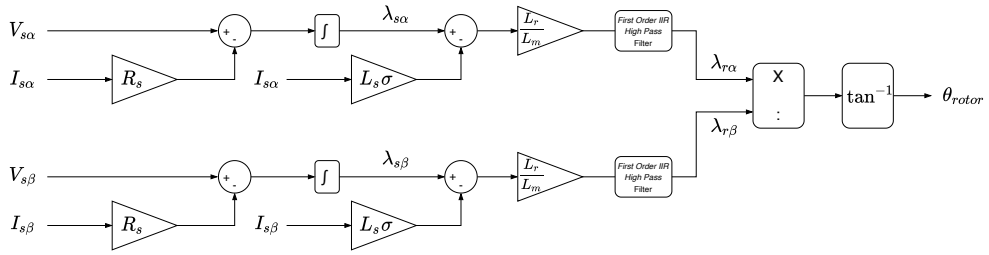


Figure 2.23: Block diagram of rotor flux observer

First-order high-pass IIR filter has the following characteristics: α is the filter coefficient and the f_{cut} is the cut-off frequency given in Figure 2.24. Since the first-order high-pass IIR filter is obtained by subtracting the first-order low-pass filter from the input signal, the filter coefficient α and cut-off frequency f_{cut} is obtained by the identical equations used for obtaining the first-order low-pass filter coefficients, which is explained in Section 2.4.16. The cut-off frequency f_{cut} is chosen as reciprocal of the rotor time constant. The Equation 2.24c is the difference equation of the first-order high-pass IIR filter. Figure 2.24 shows implementation of the first-order high-pass IIR filter, by subtracting the first-order low-pass filter response from the input signal, IIR high-pass filter is obtained. The detailed block diagram of the first-order IIR low-pass filter is shown in Figure 2.31.

$$\alpha = \left(\frac{2\pi T_s f_c}{2\pi T_s f_c + 1}\right) \tag{2.20a}$$

$$f_{cut} = \left(\frac{\alpha}{(1-\alpha)\cdot 2\pi \cdot T_s}\right) \tag{2.20b}$$

$$y(k) = (1-a)\cdot x_k - (1-a)\cdot x_{k-1} + (1-a)\cdot y_{k-1} \tag{2.20c}$$

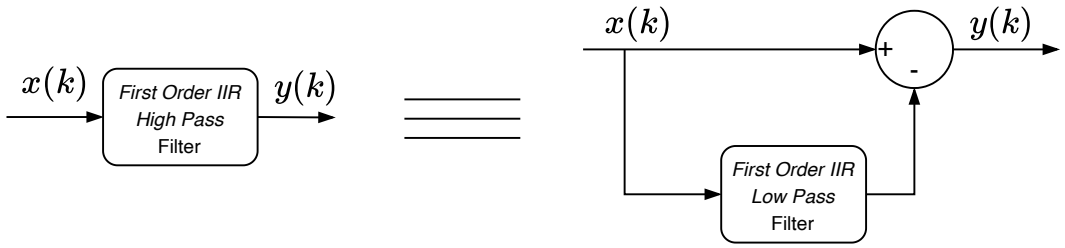


Figure 2.24: First-order IIR high-pass filter implementation

2.4.12 Dead Time Compensation

Because semiconductor switches have a finite turn-on and turn-off time, the dead time insertion is critical because if the switches on the same leg become conducted simultaneously, the DC link capacitor can become short to ground. Shoot-through occurs when a massive amount of current is released for a short time, leading the semiconductor switches to be damaged. A finite time must be inserted before turning on and off the complementary switches to prevent the shoot-through event. This finite time is known as dead time or blanking time. Duration of the dead time is related to gate capacitance of the IGBT, for the PM50RG1B120 model IPM the dead time is recommended as 2.5 μS. However, adding dead time results in non-linearity and distortion. This effect and compensation of the dead time are explained in detail in Kayhan's thesis [3].

2.4.13 Resistive Braking Module

When the motor needs to slow down from a certain speed, the rotational kinetic energy, proportional to the system's inertia and the square of the rotational speed must be transferred from the motor to another place. The motor transforms kinetic energy into electrical energy and sends it back to the inverter, this phenomenon is recognized as dynamic braking. There are different types of dynamic braking: regenerative braking and resistive braking. The energy is not dissipated in the regenerative braking; instead, it is stored in capacitors or batteries. On the other side, the energy is wasted on a resistor in the resistive braking, which is implemented in our drive system. Because the DC link capacitor's voltage ratings may be exceeded in the event of

a fast deceleration, some precautions should be taken to protect the capacitors from exceeding their voltage ratings and blowing up.

Mitsubishi's PM50RG1B120 IPM features an extra IGBT for resistive braking; we have used this IGBT for resistive braking. When the DC link voltage hits 600V, the resistive braking IGBT becomes active for 1 second, then pauses 1 second before becoming active again, just as no excess energy should be dissipated on the braking resistor.

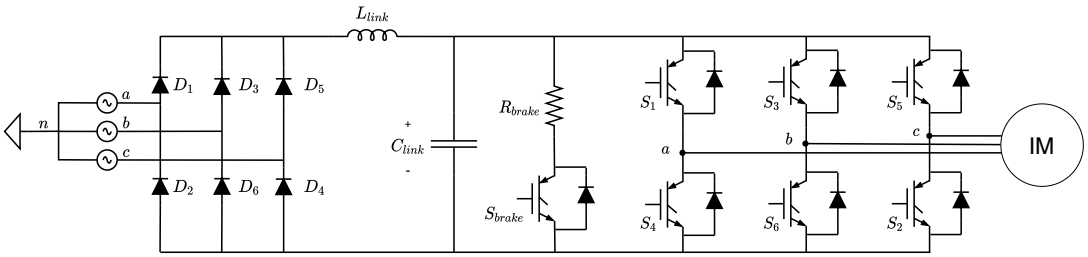


Figure 2.25: Schematic diagram of IPM with resistive braking IGBT

2.4.14 Space Vector PWM Module

Two-level voltage source inverter (VSI) and space vector pulse width modulation (SVPWM) topics are described in this section. In motor drives, the VSI is the most frequent topology. VSI converts DC voltage to variable frequency and amplitude AC voltage. VSI consist of six power transistors with six anti-parallel diodes shown in Figure 2.26.

In contrast to the sinusoidal PWM (SPWM) method, the SVPWM method offers $\%15.5$ larger fundamental voltage. The main advantages of the SVPWM are as follows [38]:

- Switching losses are less
- Harmonic distortion is smaller for the load current
- Better utilization of the DC link

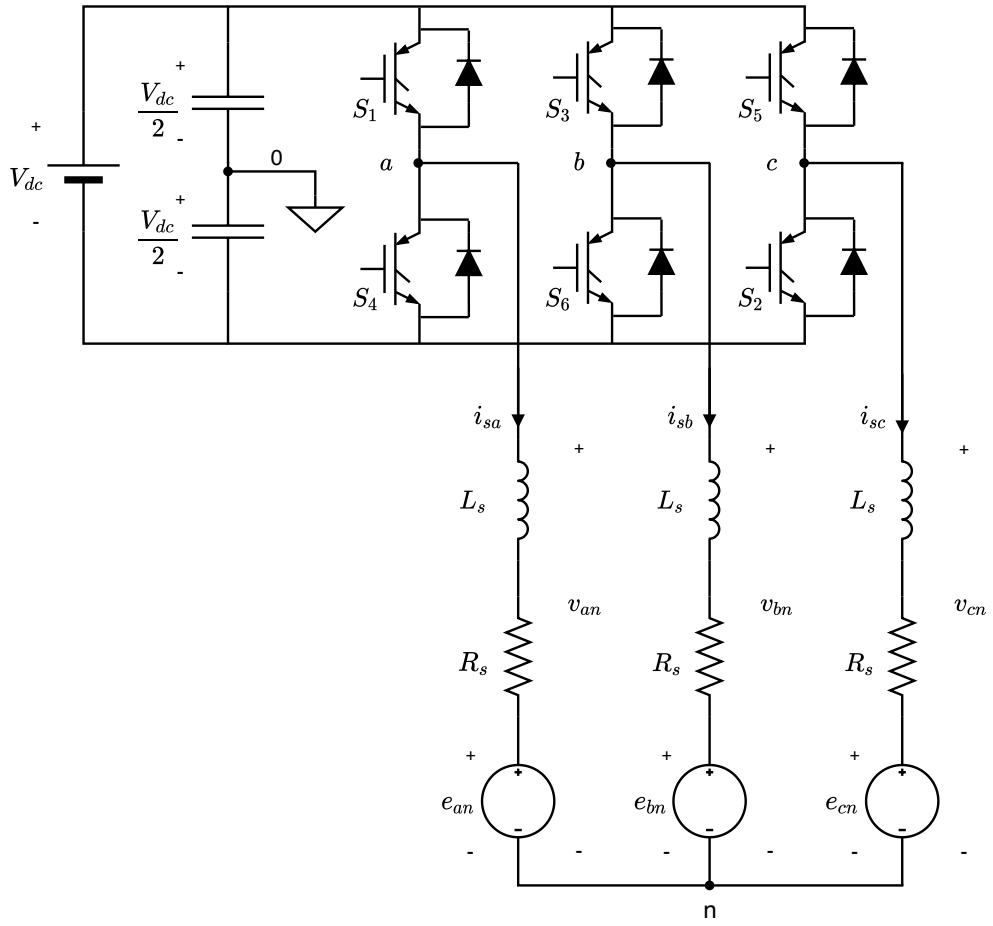


Figure 2.26: Two-level voltage source inverter topology

Figure 2.27 shows the eight possible switching states of the VSI, every state corresponds to a specific space vector shown in the graph. By combining the adjacent vectors and taking their weighted average the reference voltage can be obtained. Table 2.2 shows the phase to neutral and line to line voltages at different switch states.

There are six different sectors as shown in Figure 2.27, but we will explain only sector 1 in this chapter. Equation 2.21 shows the calculation of the on times of the switches for sector 1. K is the modulation constant, V_{dc} is the DC link voltage and T_s is the

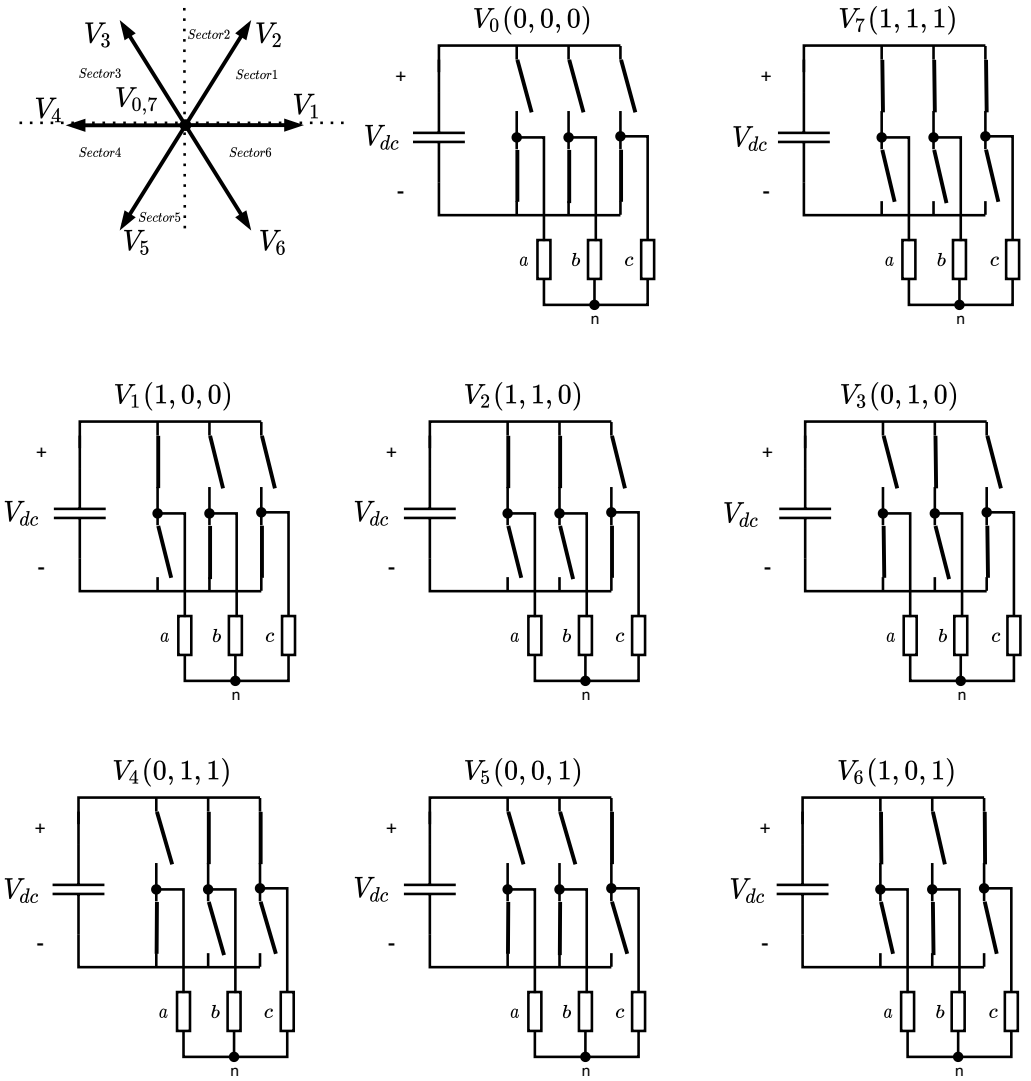


Figure 2.27: Possible switching states of VSI

sampling time.

$$K = \frac{\sqrt{3}}{2} * \frac{T_S}{V_{DC}} \tag{2.21a}$$

$$T_1 = K * \left(\sqrt{3} * V_{\text{sref}\alpha} - V_{\text{sref}\beta}\right) \tag{2.21b}$$

$$T_2 = K * V_{\text{sref}\beta} \tag{2.21c}$$

$$T_Z = T_S - (T_1 + T_2) \tag{2.21d}$$

Equation 2.22 shows the relation between the vector shown in Figure 2.28, V_{sref} is the combination of the adjacent vector and the zero vectors.

$$V_{\text{sref}} * T_S = V_0 * \frac{T_0}{2} + V_1 * T_1 + V_2 * T_2 + V_7 * \frac{T_0}{2} \tag{2.22}$$

Table 2.2: Switch states and phase to neutral and line-line voltages

Vector	S_a	S_b	S_c	v_{an}	v_{bn}	v_{cn}	v_{ab}	v_{bc}	v_{ca}
0	0	0	0	0	0	0	0	0	0
1	0	0	1	$-\frac{V_{dc}}{3}$	$-\frac{V_{dc}}{3}$	$\frac{2V_{dc}}{3}$	0	$-V_{dc}$	V_{dc}
2	0	1	0	$-\frac{V_{dc}}{3}$	$\frac{2V_{dc}}{3}$	$-\frac{V_{dc}}{3}$	$-V_{dc}$	V_{dc}	0
3	0	1	1	$-\frac{2V_{dc}}{3}$	$\frac{V_{dc}}{3}$	$\frac{V_{dc}}{3}$	$-V_{dc}$	0	V_{dc}
4	1	0	0	$\frac{2V_{dc}}{3}$	$-\frac{V_{dc}}{3}$	$-\frac{V_{dc}}{3}$	V_{dc}	0	$-V_{dc}$
5	1	0	1	$\frac{V_{dc}}{3}$	$-\frac{2V_{dc}}{3}$	$\frac{V_{dc}}{3}$	V_{dc}	$-V_{dc}$	0
6	1	1	0	$\frac{V_{dc}}{3}$	$\frac{V_{dc}}{3}$	$-\frac{2V_{dc}}{3}$	0	V_{dc}	$-V_{dc}$
7	1	1	1	0	0	0	0	0	0

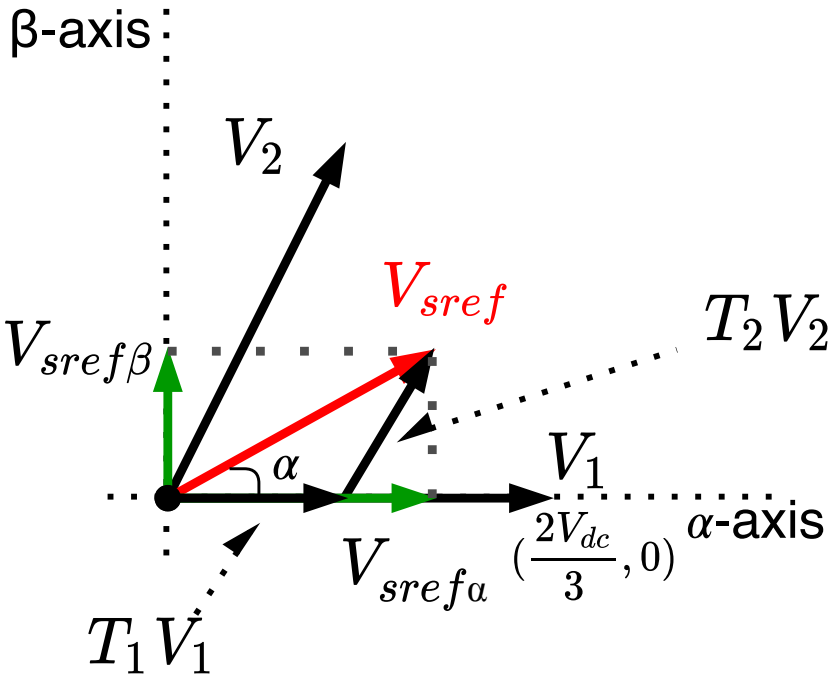


Figure 2.28: Calculation of on times

The on time calculation of the upper switches are given in Equation 2.23.

$$t_{Aon} = \frac{T_0}{2} + T_1 + T_2 \tag{2.23a}$$

$$t_{\text{Bon}} = \frac{T_0}{2} + T_1 \tag{2.23b}$$

$$t_{\text{Con}} = \frac{T_0}{2} \tag{2.23c}$$

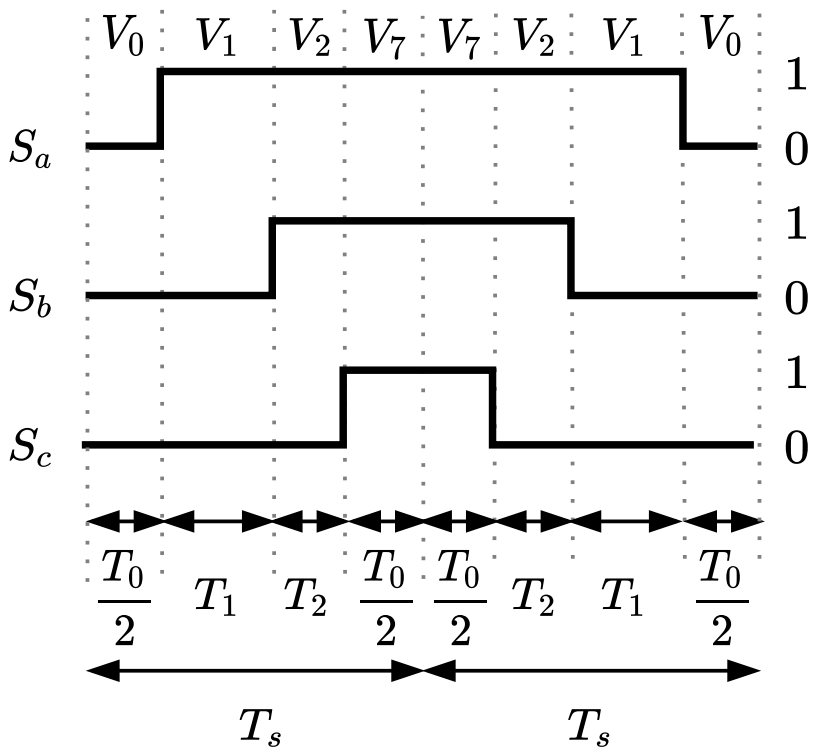


Figure 2.29: Switching sequence in sector 1

The calculation of the duty cycles are given in Equation 2.24.

$$duty_A = \frac{t_{Aon}}{T_s} \tag{2.24a}$$

$$duty_B = \frac{t_{Bon}}{T_s} \tag{2.24b}$$

$$duty_C = \frac{t_{Con}}{T_s} \tag{2.24c}$$

2.4.15 DC Link Compensation

The grid voltage (380V AC) is rectified by a three-phase diode rectifier and filtered by an inductor and a DC link capacitor in the experimental setup; the rectified voltage has an average value of 537V. The DC link ripple cannot be regulated, but it may be predicted with great accuracy.

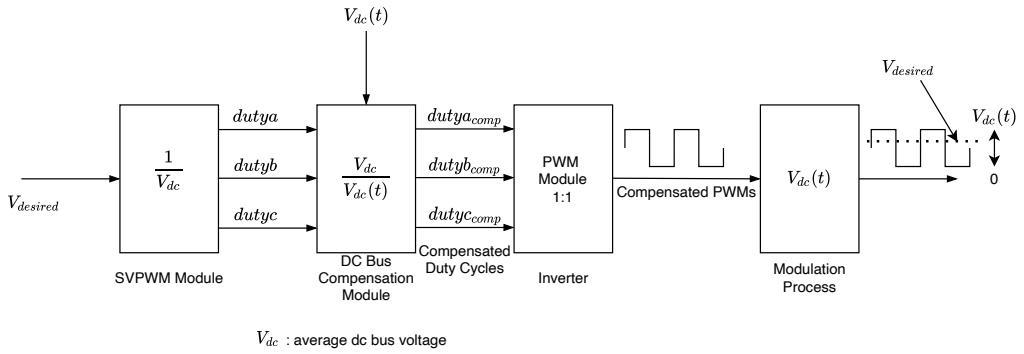


Figure 2.30: The block diagram of the DC link compensation

The duty cycles are updated according to Equation 2.25, V_{dc} represents the average DC link voltage, while $V_{dc}(t)$ represents the instantaneous DC link voltage, which is rippling 6 times the grid frequency. To eliminate the rippling effect feed-forward compensation is implemented as follows [7]:

$$dutya_{comp} = dutya * \frac{V_{dc}}{V_{dc}(t)} \tag{2.25a}$$

$$dutyb_{comp} = dutyb * \frac{V_{dc}}{V_{dc}(t)} \tag{2.25b}$$

$$dutyc_{comp} = dutyc * \frac{V_{dc}}{V_{dc}(t)} \tag{2.25c}$$

2.4.16 Speed Sensing

The rotor flux angle was calculated in Equation 2.19. Since the rotor flux is rotating and aligned with the d-axis of the synchronous reference frame, the derivative of the rotor flux angle given in Equation 2.26 equals the synchronous speed. The derivative of the angle can result in discontinuities, these can be avoided by a Matlab function block implementing Equation 2.27 in order to remove discontinuities caused by jumps from 2π to 0 [39].

$$\omega_s(k) = \frac{\theta(k) - \theta(k-1)}{T_s} \tag{2.26}$$

$$\omega_s(k) = \begin{cases} (\theta(k) - \theta(k-1) + 2\pi)/T_s & \text{if } \theta(k) - \theta(k-1) < -\pi \\ (\theta(k) - \theta(k-1) - 2\pi)/T_s & \text{if } \theta(k) - \theta(k-1) > \pi \\ (\theta(k) - \theta(k-1))/T_s & \text{otherwise.} \end{cases} \tag{2.27a}$$

(2.27b)

Often, the estimated speed value needs to be filtered. A simple filtering solution is by using an IIR low-pass filter, with the following transfer function:

$$H(z) = \frac{\alpha}{1 + (\alpha - 1)z^{-1}} \tag{2.28}$$

The α coefficient depends on the filter cutoff frequency $f_c = 1/(2\pi t_c)$ and the sampling period T_s (200 μs):

$$\alpha = \frac{T_s}{t_c + T_s} = \frac{2\pi f_c T_s}{1 + 2\pi f_c T_s} \tag{2.29}$$

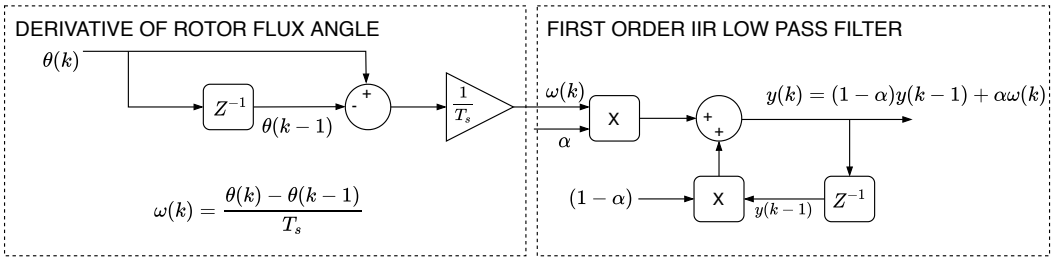


Figure 2.31: Block diagram of synchronous speed calculation

The slip speed can be derived from the circuit in Figure 2.11, the assumption is the rotor flux is aligned with the d-axis of the synchronous reference frame, Equation 2.30 shows the calculation steps of the slip speed. After finding the slip speed in Equation 2.30d, the rotor speed in Equation 2.30e is calculated by subtracting the slip speed from the synchronous speed.

$$w_{slip} * \lambda_{rq} = R_r * i_{rq} \tag{2.30a}$$

$$i_{rq} = -\frac{L_m}{L_r} i_{sq} \tag{2.30b}$$

$$\tau_r = \frac{L_r}{R_r} \tag{2.30c}$$

$$w_{slip} = s * w_s = \frac{L_m}{\tau_r \lambda_r d} i_{sq} \tag{2.30d}$$

$$w_r = w_s - s * w_s \tag{2.30e}$$

The low-pass IIR filter whose Z-transform is given in Equation 2.28a has following characteristics: α is the filter coefficient, f_{cut} is the cut-off frequency given in Equation 2.31a and Equation 2.31b respectively. The Equation 2.31c is the difference equation of the first-order low-pass IIR filter.

$$\alpha = \left(\frac{2\pi T_s f_c}{2\pi T_s f_c + 1} \right) \tag{2.31a}$$

$$f_{cut} = \left(\frac{\alpha}{(1-\alpha) \cdot 2\pi \cdot T_s} \right) \tag{2.31b}$$

$$y(k) = a \cdot x_k + (1-a) \cdot y_{k-1} \tag{2.31c}$$

Figure 2.32 shows the block diagram of the speed sensing algorithm:

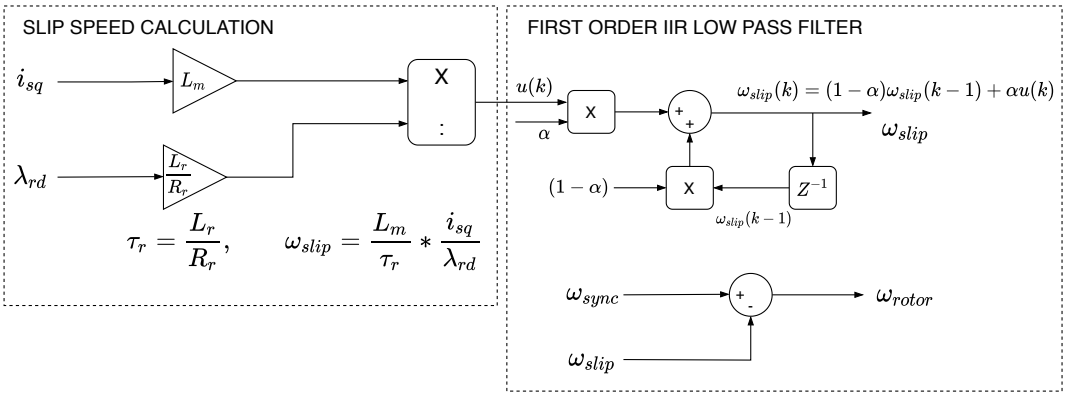


Figure 2.32: Block diagram of slip speed and rotor speed sensing

2.4.17 Speed Loop

The speed loop is cascaded to the torque control loop and it has to have at least 10 times less band-with in order not to have any effect on the current control loop. The PI controller topology is very similar to the current control loop.

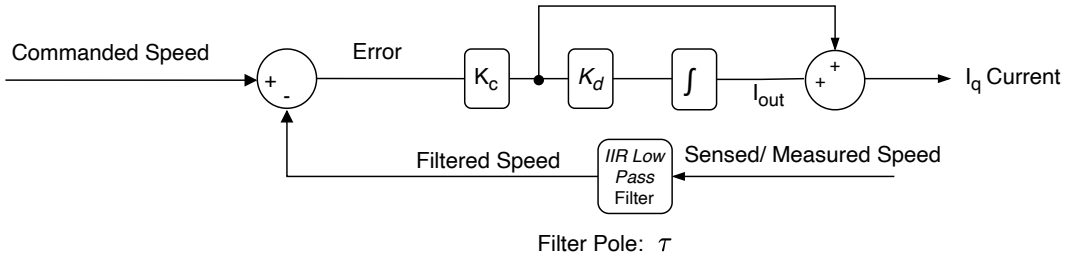


Figure 2.33: Block diagram of the speed loop PI controller [7]

K_c and K_d are calculated in Equation 2.32. P is the pole number, J is the system inertia, τ is the LPF filter pole and δ is the damping factor:

$$K = \frac{3}{4} P \frac{L_m^2}{J L_r} I_d \tag{2.32a}$$

$$K_c = \frac{1}{\delta K \tau} \tag{2.32b}$$

$$K_d = \frac{1}{\delta^2 \tau} \tag{2.32c}$$

2.4.18 PWM Generation and Dead Time Insertion Module

This module generates PWM signal from the duty cycle and inserts dead-time. Since the DS1104 inserts asymmetric dead-time to the generated PWM signals, the asymmetric dead-time insertion may move the PWM signal's center point by half of dead-time duration, resulting in voltage loss and nonlinearity.

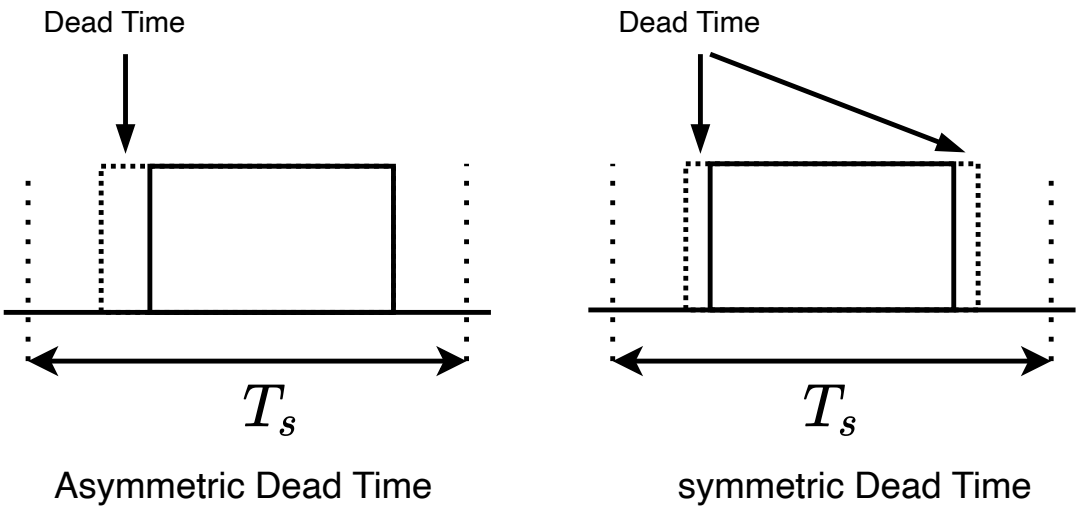


Figure 2.34: Asymmetric and symmetric dead-time insertion

2.4.19 Torque Calculation Module

The electrical torque is calculated by Equation 2.33, there are several ways to calculate the electrical torque, since we have available ψ_α , ψ_β, I_α , I_β from Section 2.4.11, we preferred to compute the electrical torque with these variables. However, the electrical torque can also be computed by the synchronous reference frame variables such as I_{ds} and I_{qs}, which is shown in Equation 2.34.

$$T_{\mathrm{e}} = \frac{3}{2} P \frac{L_m}{L_r} \left(\psi_\alpha I_\beta - \psi_\beta I_\alpha\right) \tag{2.33}$$

$$T_e = \frac{3}{2} \frac{P}{2} \frac{L_m^2}{L_r} I_{ds} I_{qs} \tag{2.34}$$

2.5 Testing the Influence of the Series PI Controller on the Torque Producing Current I_{qs}

The simulation model is designed for Lucas Nuelle's SE 2662-5H model squirrel cage induction motor. The parameters of the motor are given in Appendix A. Firstly, the simulation model is built precisely, we aimed to make improvements on the model which Kayhan has improved in his thesis [3]. Kayhan has built the vector control algorithm by S-functions which is written in C. In this thesis, the vector control algorithm is implemented by Simulink blocks and some modifications such as series PI controller and addition of axis decoupling module are done. The results were satisfactory in the simulation environment. After the model's verification in the simulation environment, the same algorithm is embedded in DSPACE DS1104. We have mainly focused on the torque bandwidth of the motor which is related to I_{qs} as shown in Equation 2.34. The vector control algorithm's accuracy is crucial for the R_s estimation algorithm implemented in the following chapters. The load disturbance response of the vector-controlled induction motor is tested by 3Nm load torque which is applied by the load induction motor controlled by Siemens Micromaster 440. The speed loop's response to 3Nm load disturbance is given in Figure 2.35:

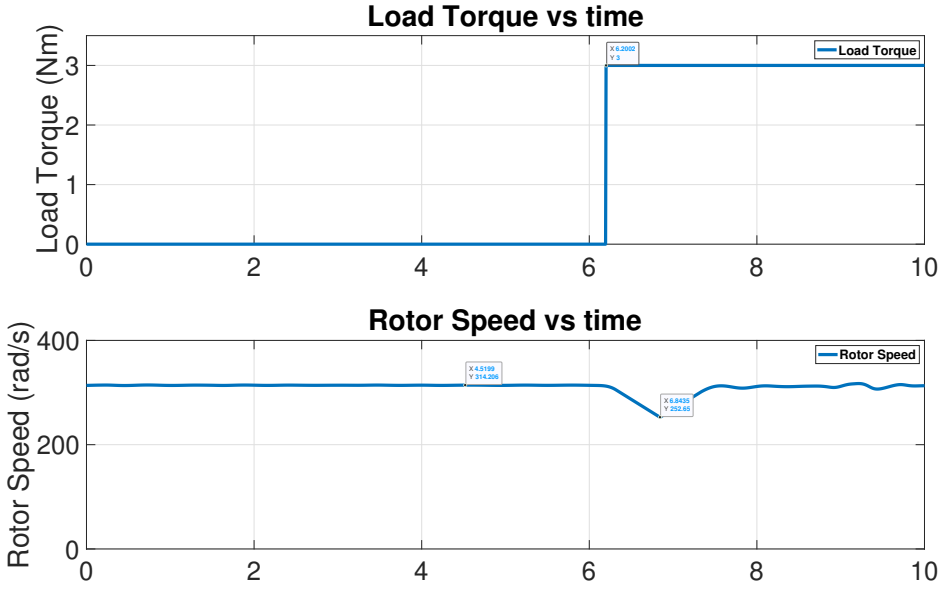


Figure 2.35: Motor speed response under 3Nm load torque

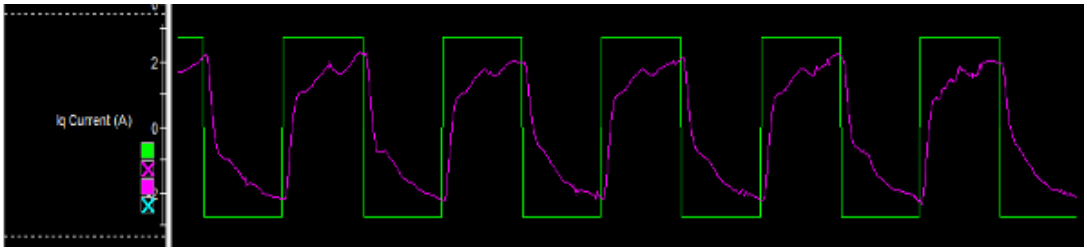


Figure 2.36: I_{qsref} vs I_{qs} from Kayhan's thesis [3]

Kayhan [3] has tested q-axis current controller's frequency response with 20 ms period square wave I_{qsref}, which has 5A peak to peak value. He got a stable response, but since he did not tuned the PI controller coefficients, the results do not reach the 2.5A peak. In Figure 2.37 and 2.38 the series PI controller is utilized and we got a oscillatory response in Figure 2.37. After changing the K_b value we got the response which is shown in Figure 2.38.

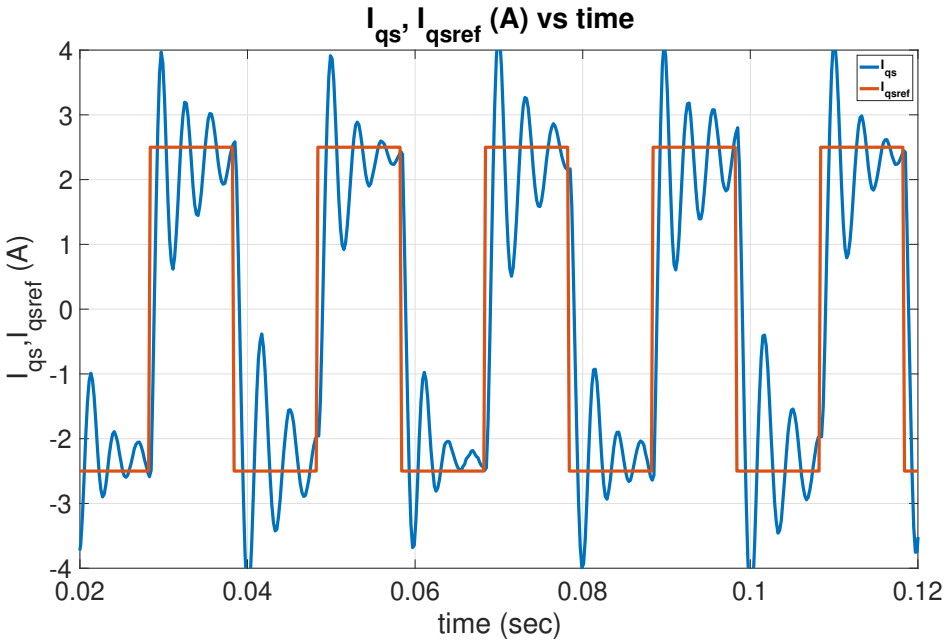


Figure 2.37: I_{qsref} vs I_{qs} for K_a=20 and K_b=400

Figure 2.37 shows the I_{qs} response for K_a=20 and K_b=400 and Figure 2.38 shows the I_{qs} response for K_a=20 and K_b=500. It can be seen that PI controller's zero location affects the I_{qs} current's frequency response. As the location of the PI controller's zero cancels the pole of the plant, we are starting to deal with a first-order system, which can be seen in Figures 2.37 and 2.38.

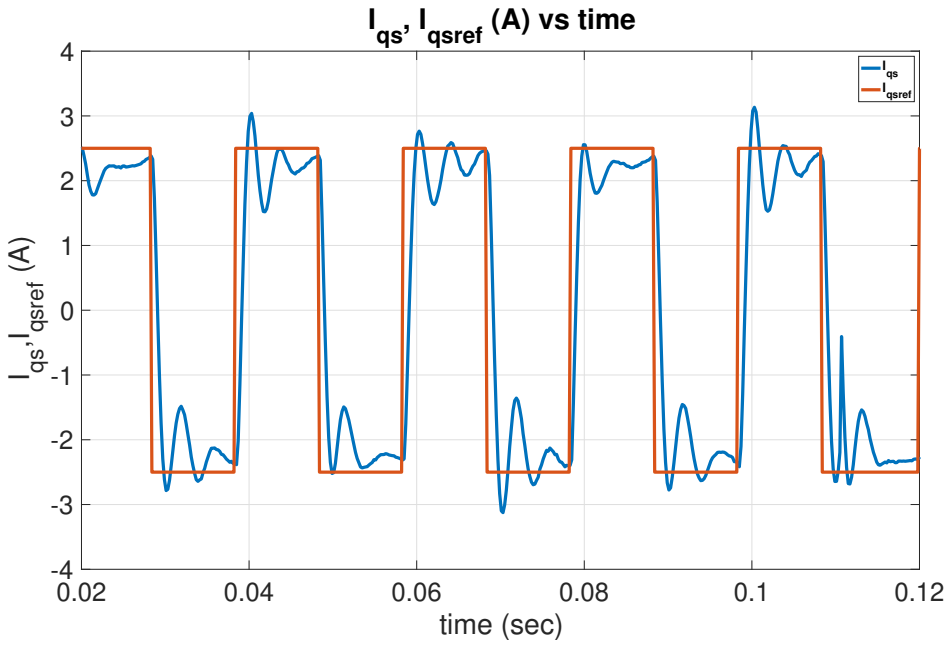


Figure 2.38: I_{qsref} vs I_{qs} for K_a=20 and K_b=500

2.6 Verification of the Model

In this section, the verification of the software is made by comparing the simulation results with the real-time experiment results. This is essential for simulating the IAEKF algorithm before it is implemented in the hardware. In this way, the possible problems can be identified at the estimation algorithm's development stage. During the tests, motor voltages, currents, power and speed of motor are measured and recorded. Table 2.3 shows the vector-controlled induction motor's speed loop activated and commanded speed is given as 3000 RPM. The same experiment is repeated for 3 Nm loaded motor running at 3000 RPM, the result of this experiment is given in Table 2.4. The simulation results and experiment results are compared to see whether the motor driver system works properly or not. From the comparisons we can conclude that the model works properly. After the verification of the model, we were able to test the IAEKF algorithm in the simulation environment.

Table 2.3: Vector-controlled induction motor's no-load test at 3000 RPM done in simulation and real-time environment

	Simulation Results	Real-Time Experiment Results
Frequency (Hz)	49,85	50,90
Voltage Line to Line Fund. (V_{rms})	380,00	379,00
Phase Current Fund. (A_{rms})	1,99	1,85
Rotor Speed (RPM)	2945,00	3029,00
Power (W)	308,00	305,00
S (VA)	880,00	780,00
Torque (Nm)	No-load	No-load
Voltage THD (%)	48,00	45,80
Current THD (%)	13,00	12,00
DC link Voltage (V)	537,00	544,00

Table 2.4: Vector controlled induction motor's 3Nm loaded test at 3000 RPM done in simulation and real-time environment

	Simulation Results	Real-Time Experiment Results
Frequency (Hz)	53,58	50,82
Voltage Line to Line Fund. (V_{rms})	380,00	387,00
Phase Current Fund. (A_{rms})	2,70	2,65
Rotor Speed (RPM)	3030,00	2875,00
Power (W)	1420,00	1345,00
S (VA)	2009,00	2061,00
Torque (Nm)	3,00	2,95
Voltage THD (%)	44,00	47,00
Current THD (%)	9,50	8,00
DC link Voltage (V)	540,00	560,00

2.7 Conclusion

This chapter implements the rotor flux-oriented vector control algorithm. To begin, the simulation model is enhanced and tested in a simulation environment. Second, the simulation model is altered and integrated into the DSPACE DS1104. The software is tested in the real-time environment. In this thesis, the execution time is chosen as 200 μs; the execution time is determined by the DSPACE DS1104's resource limits. The model developed in this chapter is critical for estimating the stator resistance achieved by the IAEKF algorithm in the following chapters. The implemented software for both simulation environment and real-time environment in Simulink are given in Appendix B. The performance comparison of the simulation model and real-time model is done in Section 2.6.

CHAPTER 3

REAL-TIME STATOR RESISTANCE ESTIMATION WITH INNOVATION BASED ADAPTIVE EXTENDED KALMAN FILTER

3.1 Introduction

In this thesis, sensorless vector control is implemented by the induction motor's voltage model and it is crucial to provide correct stator and rotor resistor values to the vector control algorithm. It is critical to know the stator resistance value to estimate the rotor's flux orientation correctly; yet, stator resistance value increases as the temperature of the motor rises. Temperature sensors, such as resistance temperature detectors (RTDs) or thermocouples are installed along the stator windings in specific motors on the market. Temperature sensors provide data that is utilized for an estimation of resistance, based on the mathematical relationship between the resistance and temperature. However, this method is not an acceptable method for a sensorless vector-controlled motor system. The research on parameter estimation started to become very popular in the '90s because the sensorless vector control was understood very well; in addition, especially in the late '80s the powerful microchips were becoming cheaper and available to the power electronics applications. In our research, we have focused on sensorless control of the induction motor. The motor's characteristics must be precisely determined in order to control the induction motor with high performance. The operational period of the induction motor can vary its parameters, and it is not always practical to determine the instant parameters of the induction motor using test equipment as stated in [40]. Since test equipment usage is not always practical for parameter estimation, estimators such as Kalman and Luenberger were studied comprehensively in the literature. An estimator is a system whose outputs are the estimate of an actual system.

Estimators can be classified as:

- Open-loop Estimators
- Closed-loop Estimators

Open-loop estimators can be described as parallel systems with the same plant matrices and inputs as the existing system. However, they do not have any error correcting operator. In Figure 3.1 the block diagram of the general open-loop observer is given.

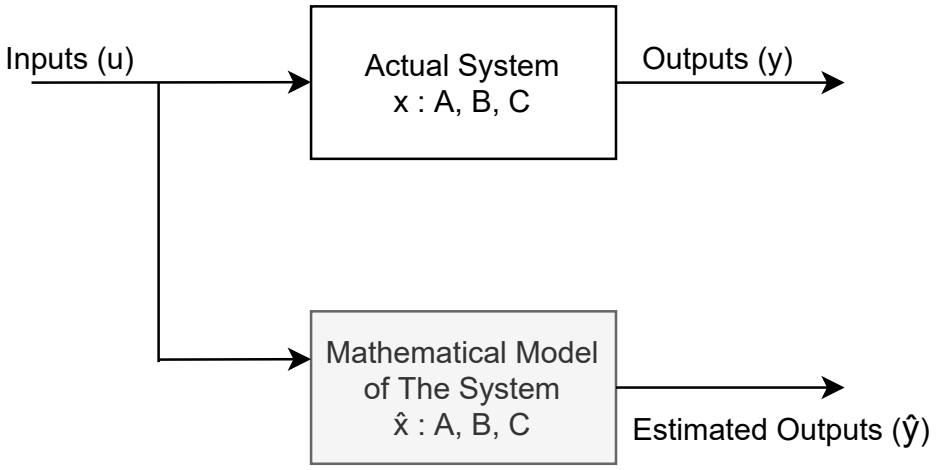


Figure 3.1: Block diagram of open-loop estimator

The most important disadvantages of open-loop estimators are that they do not utilize the output information. They will not function as intended if the system matrix is unstable or if the system is disturbed from the outside. The open-loop estimators cannot utilize the initial condition of the actual system, which makes them unreliable. On the other hand, the closed-loop observers have an error-correcting operator, and this property makes the closed-loop estimators more immune to noise, stability, and disturbance-related issues. Due to the existence of the error-correcting operator (L), the output of the observer quickly converges to the estimated parameter.

3.1.1 The Kalman Filter

In Figure 3.2 the block diagram of the generally closed-loop estimator is given. One of the commonly used closed-loop estimators is the Kalman filter. Kalman filter is

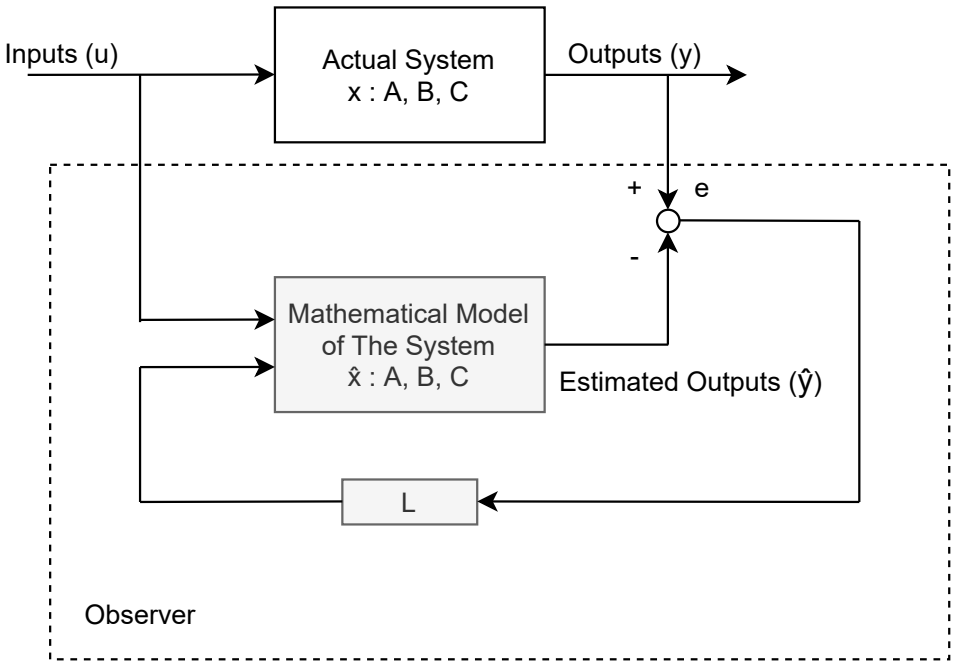


Figure 3.2: Block diagram of closed-loop estimator

not only successful at filtering the inaccurate measurements but also successful at estimating variables that cannot be measured directly by any sensor. Rudolph Kalman proposed the Kalman filter in 1960, and he made an outstanding contribution to estimation theory [41]. No system can be accurately represented in actual life, systems are typically modeled with assumptions, and dynamic equations do not include all variables. Sensors might be noisy, furthermore sensors may introduce noise to the measurements and hence affect the performance of measurement. As a result, sensors may not always deliver accurate system results. The variables and disturbances that cannot be characterized are assumed to be captured by a random vector by using Kalman filter w_k, and measurement-related uncertainties are assumed to be captured by v_k. The Kalman filter is only applicable to **stochastic time-variant linear systems** described with the following state-space equations:

$$\begin{aligned} x_{k+1} &= A_k x_k + B_k u_k + w_k \\ z_k &= H_k x_k + v_k \end{aligned} \tag{3.1}$$

Figure 3.4 shows the block diagram of the Kalman filter. $\hat{x}_k^- \in \mathbb{R}^n$ is the priori state estimate, $\hat{x}_k \in \mathbb{R}^n$ is the posteriori state estimate. $x \in \mathbb{R}^{nx1}$ is the state vec-

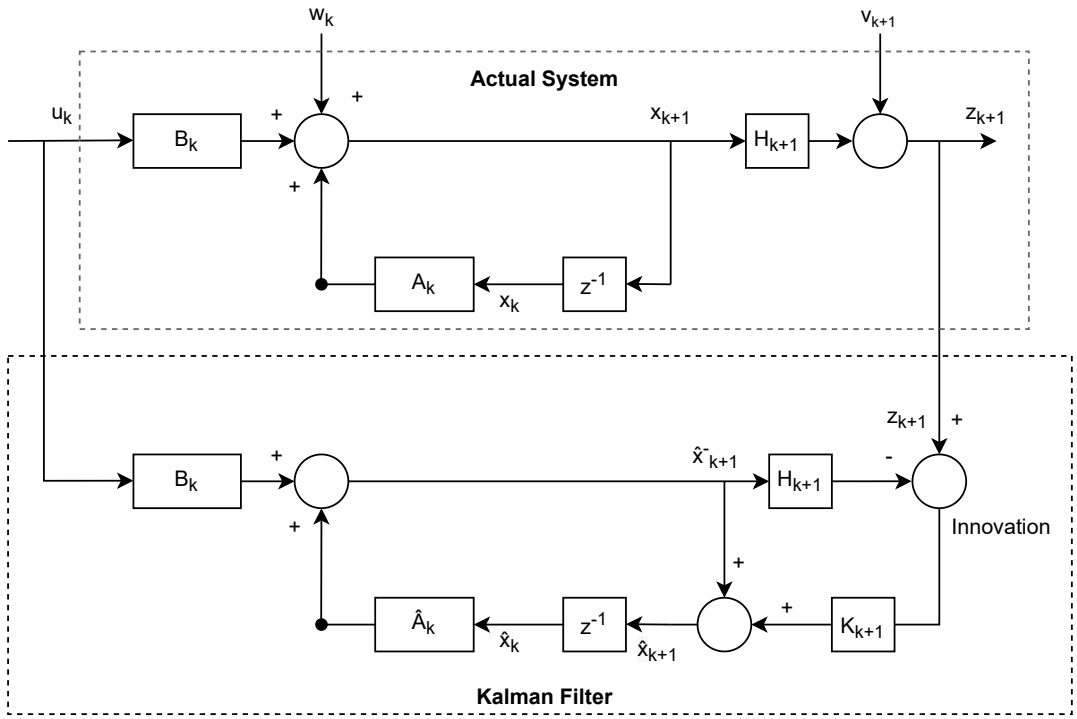


Figure 3.3: Kalman filter block diagram

tor, $u \in \mathbb{R}^{mx1}$ is the input vector and $z \in \mathbb{R}^{kx1}$ is the output vector, $A \in \mathbb{R}^{nxn}$ is the system matrix, $B \in \mathbb{R}^{nxm}$ is the input matrix, $H \in \mathbb{R}^{kxn}$ is the output matrix. This block diagram is similar to the state observer given in Figure 3.2 which does not include model measurement, and process noise; and it is used for deterministic systems; on the other hand, the Kalman filter models the measurement and process noise, and it is designed for stochastic systems. Kalman filter is a two-step algorithm, Equation 3.2 is the time update equation, the priory state estimate $\hat{x}_{k+1}^{-}$ is calculated by state equations and $\hat{P}_{k+1}^{-}$ is the variance of the priory estimate which represents the uncertainty in the priory estimate. The second step of the algorithm is measurement update equations, given in Equation 3.3, this equation gets the priori state and error the covariance estimates found in time update Equation 3.2 and it utilize former equation as input to find the Kalman gain, posteriori estimates of the state, and error covariance estimates.

$$\text{Time update} \begin{cases} \hat{x}_{k+1}^{-} = A_k \hat{x}_k + B_k u_k \\ P_{k+1}^{-} = A_k P_k A_k + Q_k \end{cases} \tag{3.2}$$

$$\text{Measurement update} \begin{cases} K_{k+1} = P_{k+1}^{-} H_k^T [H_{k+1} P_{k+1}^{-} H_{k+1}^T + R_k]^{-1} \\ \hat{x}_{k+1} = \hat{x}_{k+1}^{-} + K_{k+1}(z_{k+1} - H_{k+1}\hat{x}_{k+1}^{-}) \\ P_{k+1} = P_{k+1}^{-} - K_{k+1} H_{k+1} P_{k+1}^{-} \end{cases} \tag{3.3}$$

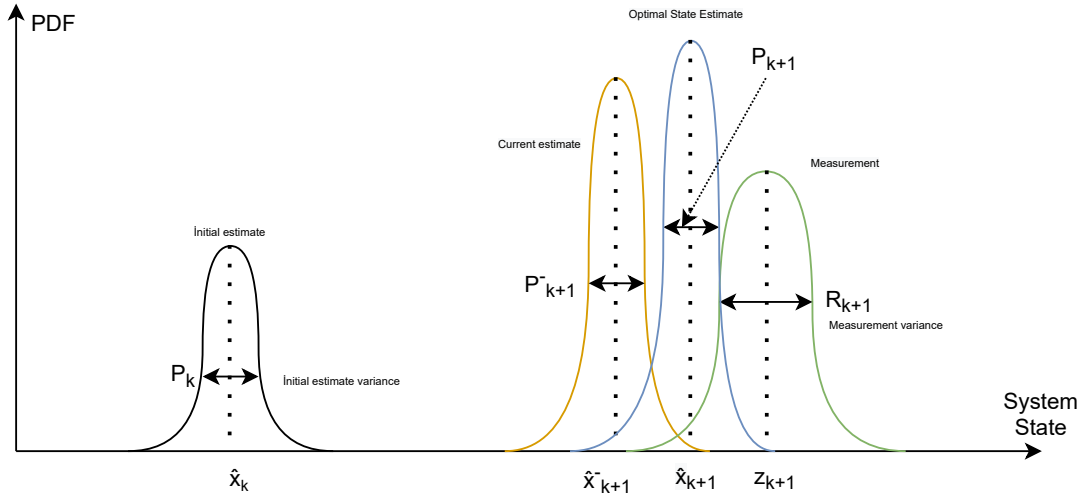


Figure 3.4: Probability density of the Kalman filter

The Kalman filter is only applicable to **stochastic time-variant linear systems**. However, the induction motors have a nonlinear model as given in Equation 3.4. The **extended Kalman filter** or **EKF** linearizes the nonlinear function around the mean of the current state estimate. In this thesis, an altered version of EKF is implemented, in Section 3.1.2 difference between the EKF and IAEKF is explained. The recursive algorithm differences between the Kalman filter and the extended Kalman filter can be seen in Figure 3.5 and Figure 3.7.

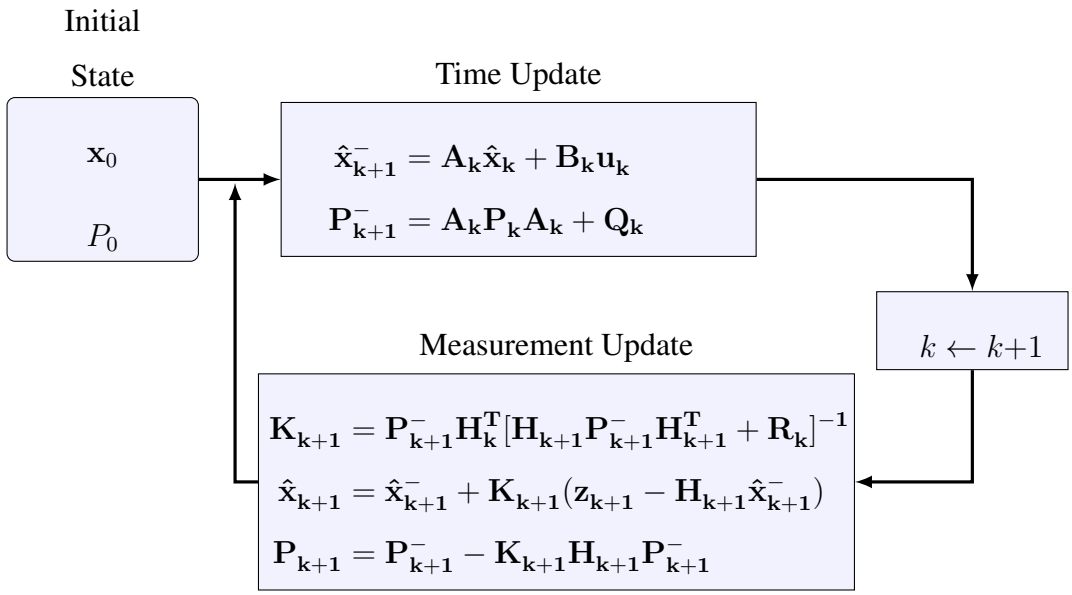


Figure 3.5: Recursive algorithm of the Kalman filter

The purpose of estimating the $\alpha\beta$ components of stator flux, $\alpha\beta$ components of stator current and stator resistance R_s is to improve the stability of the IAEKF algorithm by utilizing the correlation between the states and parameters in the nonlinear inputs applied to the estimator. In the time domain, the state space equations are obtained from the differential equations of the IM is given in Equation 3.4 and Equation 3.5.

$$
\begin{aligned}
\dot{x}_e(t) &= f_e(x_e(t), u_e(t)) + w(t) \\
\underbrace{\begin{bmatrix} \dot{i}_{s\alpha}(t) \\ \dot{i}_{s\beta}(t) \\ \dot{\varphi}_{s\alpha}(t) \\ \dot{\varphi}_{s\beta}(t) \\ \dot{R}_s(t) \end{bmatrix}}_{\dot{\mathbf{x}}_\mathbf{e}(\mathbf{t})} &= \underbrace{\begin{bmatrix} -(\frac{R_s}{L_\sigma}+\frac{R_r L_s}{L_\sigma L_r})i_{s\alpha}(t) - p_p w_m i_{s\beta}(t) + \frac{R_r}{L_r L_\sigma}\varphi_{s\alpha}(t) + \frac{p_p w_m}{L_\sigma}\varphi_{s\beta}(t) + \frac{1}{L_\sigma}v_{s\alpha}(t) \\ p_p w_m i_{s\alpha}(t) - (\frac{R_s}{L_\sigma}+\frac{R_r L_s}{L_\sigma L_r})i_{s\beta}(t) - \frac{p_p w_m}{L_\sigma}\varphi_{s\alpha}(t) + \frac{R_r}{L_r L_\sigma}\varphi_{s\beta}(t) + \frac{1}{L_\sigma}v_{s\beta}(t) \\ -R_s i_{s\alpha}(t) \\ -R_s i_{s\beta}(t) \\ 0 \end{bmatrix}}_{\mathbf{f_e}(\mathbf{x_e}(\mathbf{t}),\mathbf{u_e}(\mathbf{t}))} \\
&\quad + w_1(t)
\end{aligned}
\tag{3.4}
$$

Output variables:

$$
\begin{aligned}
y_e(t) &= h_e(x_e(t)) + w_2(t) \\
&= \underbrace{\begin{bmatrix} i_{s\alpha}(t) \\ i_{s\beta}(t) \end{bmatrix}}_{\mathbf{h_e(x_e(t))}} + w_2(t) \\
&= \underbrace{\begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \end{bmatrix}}_{\mathbf{h_e}} \underbrace{\begin{bmatrix} i_{s\alpha} \\ i_{s\beta} \\ \varphi_{s\alpha} \\ \varphi_{s\beta} \\ R_s \end{bmatrix}}_{\mathbf{x_e}} + w_2(t)
\end{aligned}
\tag{3.5}
$$

In Equation 3.4 and Equation 3.5 the $x_e \in \mathbb{R}^n$ represents system state vector of the induction motor, $f_e(x_e(t), u_e(t)) \in \mathbb{R}^n$ is the nonlinear function of the input and the system state vector, $h_e(x_e(t)) \in \mathbb{R}^k$ is the measurement function, $y_e(t) \in \mathbb{R}^k$ is the measurement output, $w_1(t) \in \mathbb{R}^n$ is the process noise and the $w_2(t) \in \mathbb{R}^k$ is the measurement noise of the output function. To be able to apply IAEKF algorithm, the nonlinear system matrix has to be transformed into discrete and linear form. To discretize the $f_e(x_e(t), u_e(t))$ function, forward Euler discretization method with the sample time of T is utilized.

The coefficients L_σ and σ in Equation 3.4 are defined as $L_\sigma = \sigma L_s$ is the stator transient inductance and $\sigma = 1 - \frac{L_m^2}{L_s L_r}$ is the leakage factor. Since the EKF algorithm is implemented for discrete systems, the continuous state-space equations are discretized by the forward Euler method with the sample time of T; the following equations achieve the discretization in Equation 3.9,

$$
\begin{aligned}
\dot{x}(t) &= Ax(t) + Bu(t) \\
y(t) &= hx(t)
\end{aligned}
\tag{3.6}
$$

Discretization of Equation 3.6 with very small T gives the following discrete state space equation:

$$
\begin{aligned}
\frac{x(k+1) - x(k)}{T} &= Ax(k) + Bu(k) \\
x(k+1) &= (I + AT)x(k) + Bu(k)T
\end{aligned}
\tag{3.7}
$$

$$y(k) = hx(k) \tag{3.8}$$

In summary the relation between the continuous state space and discrete state space equation state matrices are given as:

$$\begin{aligned} A &\approx I + AT \\ B &\approx BT \\ h &\approx h \end{aligned} \tag{3.9}$$

From Equation 3.7 it can be seen that the discrete-time state-space model's accuracy depends on the sample time; as the sample time decreases, the discrete state-space model reflects the properties more accurately. In this thesis, the sample time is chosen as 200 μs; this sample time is determined by the DSPACE DS1104's resource limits. The EKF model has a very high computational cost since the Jacobian matrices are computed numerically. Discrete state-space model of induction motor is obtained after discretizing the continuous state-space model with Equation 3.9. The discrete state-space equation of the induction motor is given in Equation 3.10:

$$\underbrace{\begin{bmatrix} i_{s\alpha,k+1} \\ i_{s\beta,k+1} \\ \phi_{s\alpha,k+1} \\ \phi_{s\beta,k+1} \\ R_{s,k+1} \end{bmatrix}}_{\mathbf{x_{k+1}}} = \underbrace{\begin{bmatrix} (1\text{-}\frac{R_{s,k}T}{L_\sigma}\text{-}\frac{R_r L_s}{L_\sigma L_r})i_{s\alpha,k}\text{-}p_p w_{m,k} T i_{s\beta,k}+\frac{R_r T \phi_{s\alpha,k}}{L_r L_\sigma}+\frac{p_p w_{m,k} T \phi_{s\beta,k}}{L_\sigma}+\frac{T v_{s\alpha,k}}{L_\sigma} \\ p_p w_{m,k} T i_{s\alpha,k}+(1\text{-}\frac{R_{s,k}T}{L_\sigma}\text{-}\frac{R_r L_s T}{L_\sigma L_r})i_{s\beta,k}+\frac{-p_p w_{m,k} T \phi_{s\alpha,k}}{L_\sigma}+\frac{R_r T \phi_{s\beta,k}}{L_r L_\sigma}+\frac{T v_{s\beta,k}}{L_\sigma} \\ (1\text{-}R_{s,k}T)i_{s\alpha,k} \\ (1\text{-}R_{s,k}T)i_{s\beta,k} \\ R_{s,k} \end{bmatrix}}_{\mathbf{f_e(x_e(t),u_e(t))}}$$

$$+ \mathbf{w_{1,k}}$$

(3.10)

The discrete model obtained in Equation 3.10 have to be linearized by Taylor series expansion given in Equation 3.11 and $f_e(x_e(t), u_e(t)) = A_e x_e + B_e u_e$ becomes linear around the mean of the state estimate.

$$\begin{aligned} A(x_k) &= \frac{\partial f_e(x_k, u_k)}{\partial x_k}\Big|_{x_k=\hat{x}_k} \\ B &= \frac{\partial f_e(x_k, u_k)}{\partial u_k}\Big|_{u_k=\hat{u}_k} \\ H(x_k) &= \frac{\partial h(x_k)}{\partial x_k}\Big|_{x_k=\hat{x}_k} \end{aligned} \tag{3.11}$$

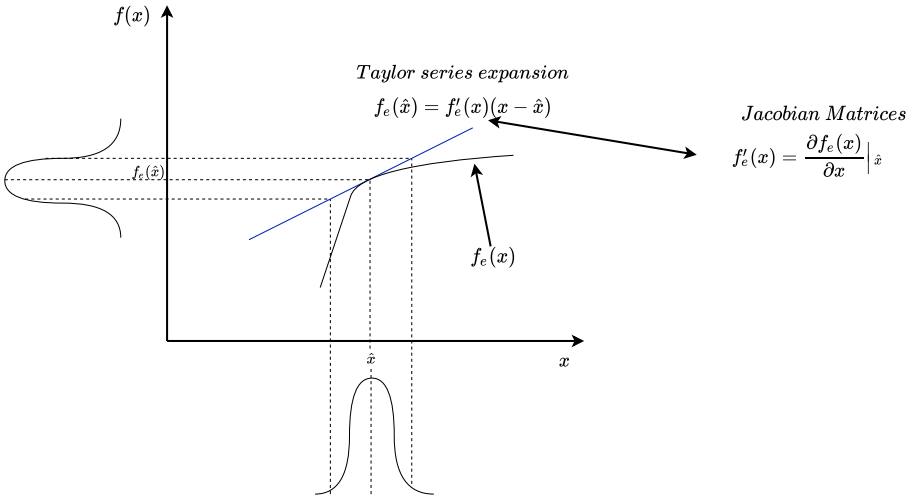


Figure 3.6: Taylor expansion of a nonlinear function

The discrete time state space representation of the model:

$$\mathbf{x}_{k+1} = \mathbf{f}_e\left(\mathbf{x}_k, \mathbf{u}_k\right) + \mathbf{w}_1 \tag{3.12a}$$

$$= \mathbf{A}\left(\mathbf{x}_k\right)\mathbf{x}_k + \mathbf{B}\mathbf{u}_k + \mathbf{w}_1 \tag{3.12b}$$

$$\mathbf{z}_k = \mathbf{h}\left(\mathbf{x}_k\right) + \mathbf{w}_2 \text{ (Measurement Equation)} \tag{3.12c}$$

$$= \mathbf{H}_k\mathbf{x}_k + \mathbf{w}_2 \tag{3.12d}$$

In the above equations x_k is the system state vector of the induction motor, f_e is the nonlinear function of system state vector x_k and u_k, u_k is the input vector, A is the system matrix, B is the input matrix of the system. H_k is the measurement vector, while h is the measurement vector. Lastly, w_1 is the process noise, and w_2 is the measurement noise. The matrix equations of the fifth-order stator flux-based

discretized and linearized induction motor model is detailed below in Equation 3.13 :

$$
\begin{gathered}
\boldsymbol{x}_k = \begin{bmatrix} i_{s\alpha,k} & i_{s\beta,k} & \varphi_{s\alpha,k} & \varphi_{s\beta,k} & R_{s,k} \end{bmatrix}^T \\
\mathbf{A}\left(\mathbf{x}_k\right) = \begin{bmatrix} a_1 & -a_2 & a_3 & a_4 & 0 \\ a_2 & a_1 & -a_4 & a_3 & 0 \\ -R_{s,k}T & 0 & 1 & 0 & 0 \\ 0 & -R_{s,k}T & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}, \mathbf{B} = \begin{bmatrix} b_1 & 0 \\ 0 & b_1 \\ T & 0 \\ 0 & T \\ 0 & 0 \end{bmatrix} \\
\mathbf{u}_k = \begin{bmatrix} v_{s\alpha,k} \\ v_{s\beta,k} \end{bmatrix}, \mathbf{z}_k = \begin{bmatrix} i_{s\alpha,k} \\ i_{s\beta,k} \end{bmatrix}, \mathbf{H} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \end{bmatrix}
\end{gathered}
\tag{3.13}
$$

In Equation 3.13 coefficients of $A\left(x_k\right)$ and B are: $a_1 = 1 - \left(R_{s,k}/L_\sigma + R_r L_s/L_r L_\sigma\right) T$, $a_2 = p_p \omega_{m,k} T$, $a_3 = R_r T/L_r L_\sigma$, $a_4 = p_p \omega_{m,k} T/L_\sigma$, $b_1 = T/L_\sigma$. The stationary reference frame components of stator voltages are $v_{s\alpha}$ and $v_{s\beta}$. Pole pair is given as p_p. L_s is the stator self-inductance. R_r is the rotor resistance. L_r is the rotor self-inductance. L_m is the mutual inductance. $L_\sigma = \sigma L_s$ is the stator transient inductance. $\sigma = 1 - L_m^2/L_s L_r$ is the leakage factor [42], [43].

3.1.2 IAEKF Design

In this thesis the IAEKF estimator is achieved by the fifth order stator flux-based model of the induction motor to estimate $i_{s\alpha}$, $i_{s\beta}$, $\varphi_{s\alpha}$, $\varphi_{s\beta}$ and R_s. IAEKF has different properties in comparison to the conventional EKF algorithm, to understand these differences the EKF algorithm is

$$\text{Linearization} \begin{cases} F_{k+1|k} = \dfrac{\partial f_e(x_k, u_k)}{\partial x_k}\Big|_{x_k=\hat{x}_k} \end{cases} \tag{3.14}$$

$$\text{Time update} \begin{cases} \hat{x}_{k+1}^{-} = f_e(\hat{x}_k, u_{k+1}) \\ P_{k+1}^{-} = F_{k+1|k} P_k F_{k+1|k}^T + Q_k \end{cases} \tag{3.15}$$

$$\text{Measurement update} \begin{cases} K_{k+1} = P_{k+1}^{-} H_k^T [H_k P_{k+1}^{-} H_k^T + R_k]^{-1} \\ \hat{x}_{k+1} = \hat{x}_{k+1}^{-} + K_{k+1}(z_{k+1} - H_k \hat{x}_{k+1}^{-}) \\ P_{k+1} = P_{k+1}^{-} - K_{k+1} H_k P_{k+1}^{-} \end{cases} \tag{3.16}$$

in the above equations, F is the linear function of the states, H_k is the linear function of the measurement equation. P_{k+1}^{-} is the extrapolation error matrix and P_k estimation error matrix. K_{k+1} is the Kalman gain. $\hat{x}_{k+1}^{-}$ is the predicted state matrix and $\hat{x}_{k+1}$ is the estimation matrix of the states. Q_k is the process error and R_k is the measurement error.

$$\text{Q equations update} \begin{cases} \hat{C}_{vk} = \sum_{j=j_0}^{k} v_j v_j^T \\ v_k = z_k - H_k \hat{x}_{k+1} \\ \hat{Q}_k = K_{k+1} \hat{C}_{vk} K_{k+1}^T \end{cases} \tag{3.17}$$

In the IAEKF algorithm, the R matrix is calculated as in the proposed method in [44] and the Q matrix is calculated by Equation 3.17. The calculation of the Q matrix is the main difference between IAEKF and the conventional EKF. The covariance estimation of the innovation residual is calculated by $\hat{C}_{vk} = \sum_{j=j_0}^{k} v_j v_j^T$, where $J_o = k - N + 1$ is the first sample of the sequence in the estimation window. V_k is the innovation residual in the Kalman filter and it is defined as $v_k = z_k - H_k \hat{x}_{k+1}$.The estimation of the process noise is obtained as in the last Equation of 3.17 in terms of the

innovation sequence under the assumption of steady-state. Unlike method proposed in reference [35], in Equation 3.17 only the diagonal elements of $\hat{Q}_k$ are determined.

3.1.2.1 Calculation of the Measurement Covariance Matrix

The calculation of the noise covariance matrix is done by measuring the phase currents of the induction motor. The current sensors are LEM company's LA55-P/SP1 current sensors. Since the current sensors and analog amplifier circuits are the same, their measurement and noise characteristics are assumed to be identical. Therefore R_k is chosen as $\sigma_R^2 x I_{2x2}$. The stochastic noise of the current sensor and analog amplifier's circuit and the quantization of analog to digital converter (ADC) influence the measurement. To determine the diagonal elements of the R_k, the standard deviation of the measured data and the quantization error of the 12-bit ADC of the DSPACE DS1104 are calculated.

$$\sigma_{ADC} = \frac{I_{LA55Pmax}}{\sqrt{3}x2^{12}} = \frac{100A}{2^{12}} = 24.41mA \tag{3.18}$$

The value of the $\sigma_{transducer}$ is calculated by taking the standard deviation of the measured data, which is obtained when only the motor is not running, and only the sensors are on. The standard deviation of the $\sigma_{transducer} = 3.48$ mA is found [45].

$$\sigma_R^2 = \sigma_{transducer}^2 + \sigma_{ADC}^2 \cong 6.09x10^{-4}A^2 \tag{3.19}$$

$$R_k(1,1) = R_k(2,2) = \sigma_R^2 \tag{3.20}$$

As it can be seen in Figure 3.7, time and measurement updates repeated recursively.

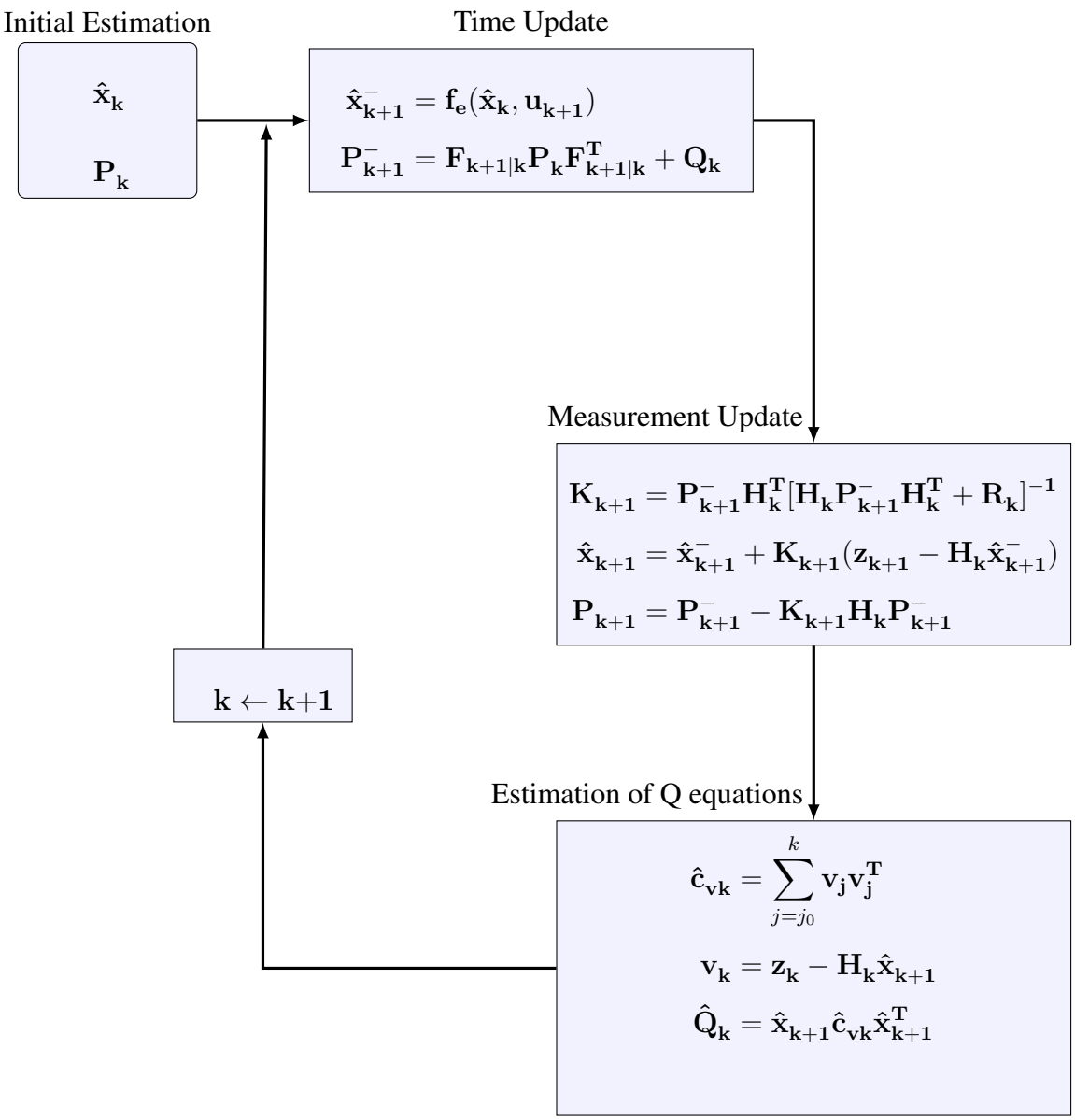


Figure 3.7: Recursive algorithm of the innovation based extended Kalman filter

3.1.3 Conclusion

In this chapter the theory of the Kalman filter and IAEKF algorithm are explained. In this thesis, the induction motor's stator resistance estimation is achieved on simulation environment for running and standstill conditions. Simulation model and simulation results can be found in Chapter 4. Then the same model is embedded into DSPACE DS1104 to verify the algorithm in the real-time environment by using our custom experimental setup. The experiment results for standstill tests can be found in Chapter 5 and the experiment results for running tests can be found in Chapter 6.

CHAPTER 4

SIMULATION MODEL OF THE IEAKF ALGORITHM

4.1 Simulink Model of the IAEKF

The IAEKF algorithm is implemented in the Matlab® Simulink environment, with the same sample time and solver configuration embedded in the DSPACE DS1104 controller board. The Simulink model is given in Figure 4.1. Figure 4.2 shows the details of IAEKF algorithm's Simulink model. We have simulated this model before embedding it into DS1104 DSP to ensure the model's accuracy. We aimed to investigate the parameters important for accurate estimation in the simulation. We have changed the window length N from 4 to 64 to see the effect, which is shown in Table 4.1. Then we have changed the calculated diagonal elements of noise covariance matrix $R_k[A]$, which is shown in Table 4.2. Since the minimum execution time of the DSPACE DS1104 is 200 µs, the execution time of simulation is also set to 200 µs.

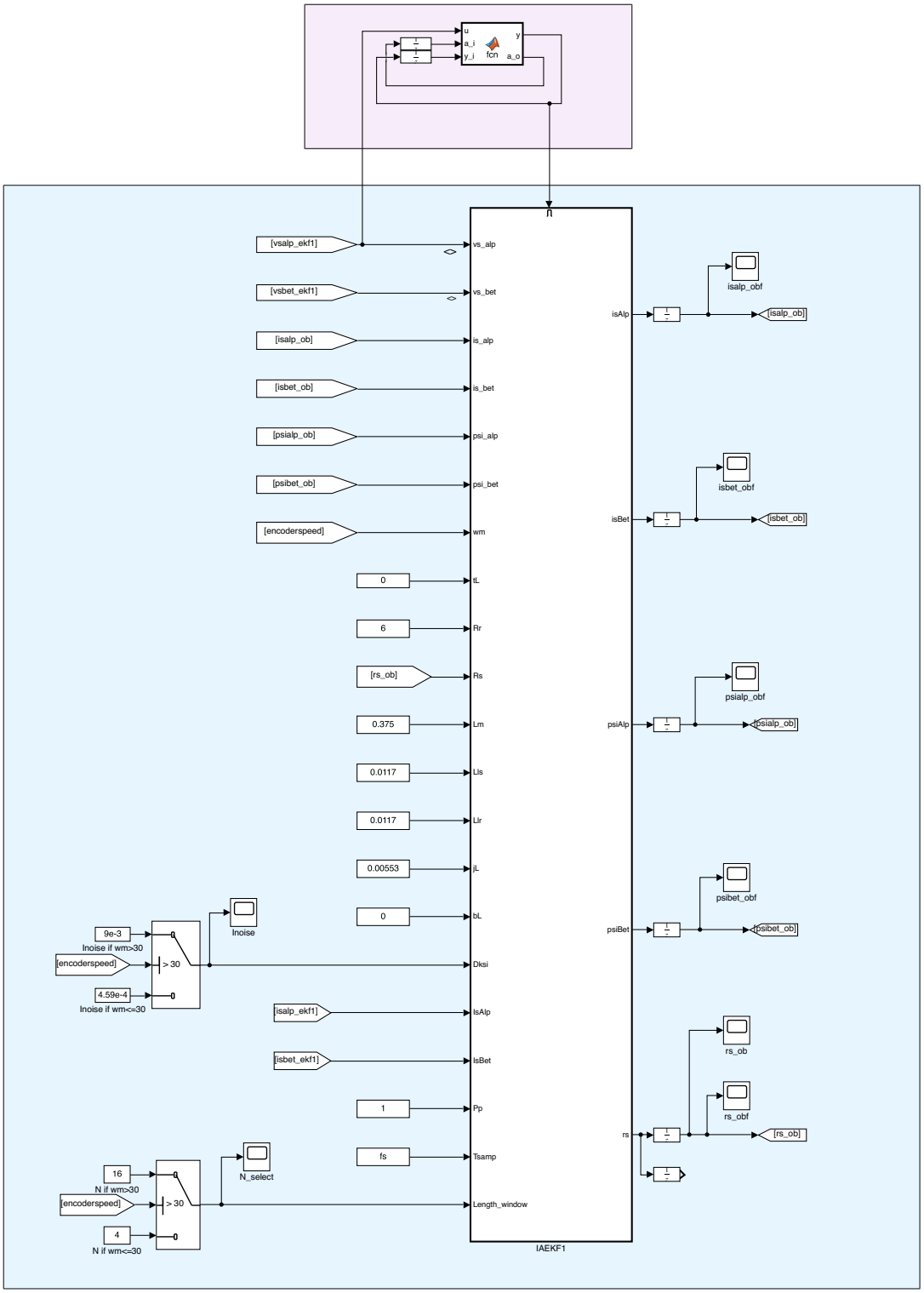


Figure 4.1: IAEKF algorithm's subsystem in Simulink

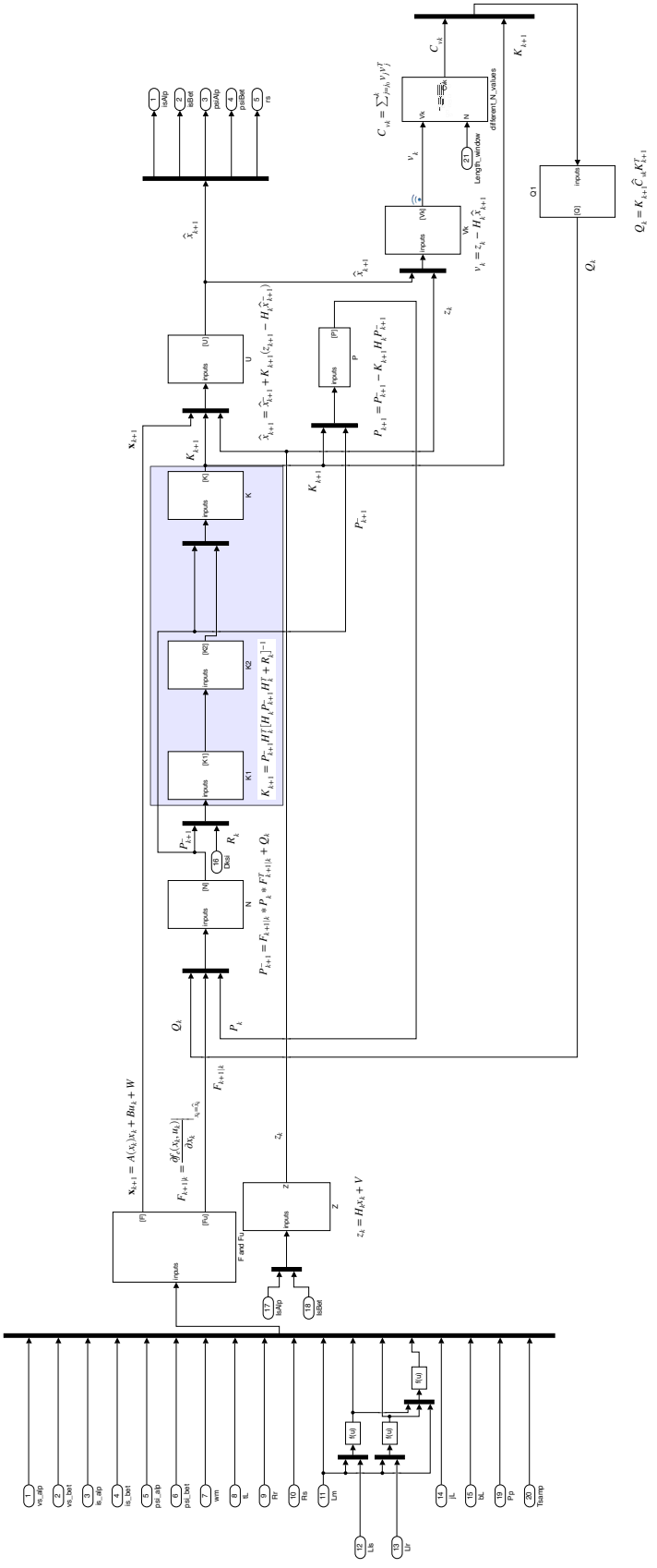


Figure 4.2: IAEKF algorithm's implementation in Simulink

Figure 4.2 shows the IAEKF algorithm's internal blocks and the equations implemented by these blocks. The explanation of these blocks are as follows:

- In the block F and F_u, nonlinear and linear functions of the inputs are calculated. Here, F_u matrix is obtained from F matrix by linearizing this function with Taylor Series around the equilibrium point. The F portion of the block is implementing the equation $x_{k+1} = A(x_k)x_k + Bu_k + W$ while the F_u portion of the block is implementing the equation $F_{k+1|k} = \frac{\partial f_e(x_k,u_k)}{\partial x_k}\Big|_{x_k=\hat{x}_k}$.

- In the block Z, measurement matrix is generated by using the measured -$\alpha\beta$ components of stator currents. This function is implementing the following equation: $z_k = H_k x_k + V$.

- In the block N, extrapolation errors matrix is calculated. This block is implementing the function $P^-_{k+1} = F_{k+1|k}P_k F^T_{k+1|k} + Q_k$.

- In these blocks (K1, K2, and K), the Kalman gain is calculated with the equation $K_{k+1} = P^-_{k+1}H^T_k[H_k P^-_{k+1}H^T_k + R_k]^{-1}$. These blocks spend a lot of computing source.

- In the block P, the estimation error matrix is calculated with the equation $P_{k+1} = P^-_{k+1} - K_{k+1}H_k P^-_{k+1}$.

- In the block U, estimation matrix of the states parameters are calculated with equation $\hat{x}_{k+1} = \hat{x}^-_{k+1} + K_{k+1}(z_{k+1} - H_k\hat{x}^-_{k+1})$.

- In the block V_k, innovation residual in the Kalman filter is calculated in this block with the equation $v_k = z_k - H_k\hat{x}_{k+1}$.

- In the block $C_v k$, the estimation of the covariance of the innovation residual is obtained by averaging the previous residual sequences for a window of the length of N (sliding window width). The equation implemented in this block is $C_{vk} = \sum_{j=j_0}^{k} v_j v_j^T$.

- In the block Q, the estimation of the process noise is implemented with equation $Q_k = K_{k+1}\hat{C}_{vk}K^T_{k+1}$.

4.2 Simulation Results

The simulations are done for offline and online estimations. The first simulation is done in standstill condition, the second simulation is done in the running condition (rated speed). To test the success of resistance detection of the algorithm 1 Ohm series resistance is added in the simulation. All of the tests are done at 50Hz frequency. Since in the simulation environment the skin effect and proximity effect can not modelled there is not any frequency-dependent result. We also tested the IAEKF model's performance for different window length N values and different elements of noise covariance matrix values in Table 4.1 and Table 4.2.

Online simulations are repeated for different values of noise covariance matrix $R_k[A]$ values as shown in Table 4.1. The algorithm is not sensitive to changes in noise covariance values up to one third of the calculated value.

Table 4.1: Estimated $\hat{R}_s$ for different elements of noise covariance matrix $R_k[A]$ values

$R_k[A]$	0.000230	0.000321	0.000413	0.000459	0.000505	0.000597	0.000689
$\hat{R}_s$ (Ω)	4.498	4.498	4.498	4.498	4.498	4.498	4.498

Online simulations are repeated for different window length N values to observe the effect as shown in Table 4.2. It can be observed that even a low value of window length N=4 is enough to estimate the stator resistance successfully.

Table 4.2: Estimated $\hat{R}_s$ for different window length N values

N	4	8	16	32	64	128	256	512
$\hat{R}_s$ (Ω)	4.498	4.498	4.498	4.498	4.498	4.498	4.498	4.498

Figure 4.3 shows stator resistance estimation for a starting value of $0\ \Omega$ and Figure 4.4 shows stator resistance estimation for a starting value of $4.45\ \Omega$. These Figures illustrates that the initial value affects the convergence time until reaching to the estimated value, however the estimated values are identical after the steady state is reached.

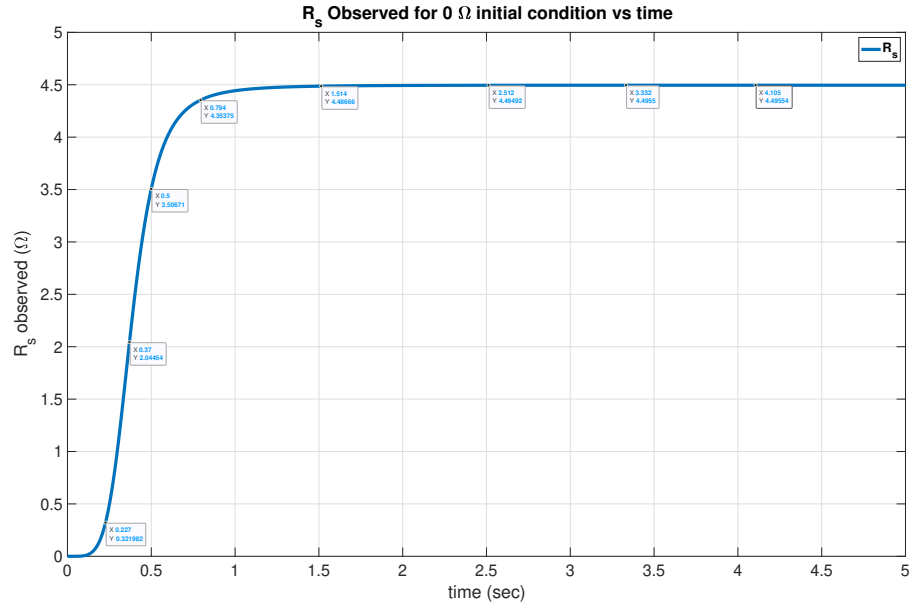


Figure 4.3: Convergence of the estimator when the initial condition for $\hat{R}_s$ 0 Ω

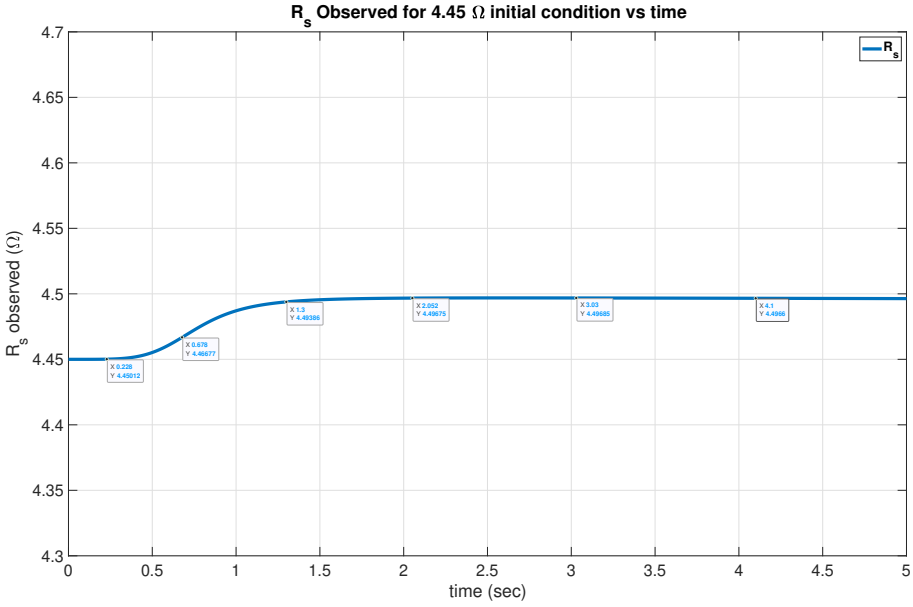


Figure 4.4: Convergence of the estimator when the initial condition for $\hat{R}_s$ is 4.45 Ω

4.2.1 Stator Resistance Estimation for Standstill Condition

This simulation is done while the motor is at standstill. The phase voltage of the motor has a magnitude of 19.5 V_{peak} and frequency of 50Hz. The window length is chosen

as N=4 and the diagonal elements of noise covariance matrix $R_k[A]$ value is chosen as $\sigma = 0.000459$. Figure 4.5 shows the simulation result for the stator resistance value $R_s = 4.45\,\Omega$. Figure 4.6 shows the simulation result for stator resistance with an additional series $1\,\Omega$ resistance.

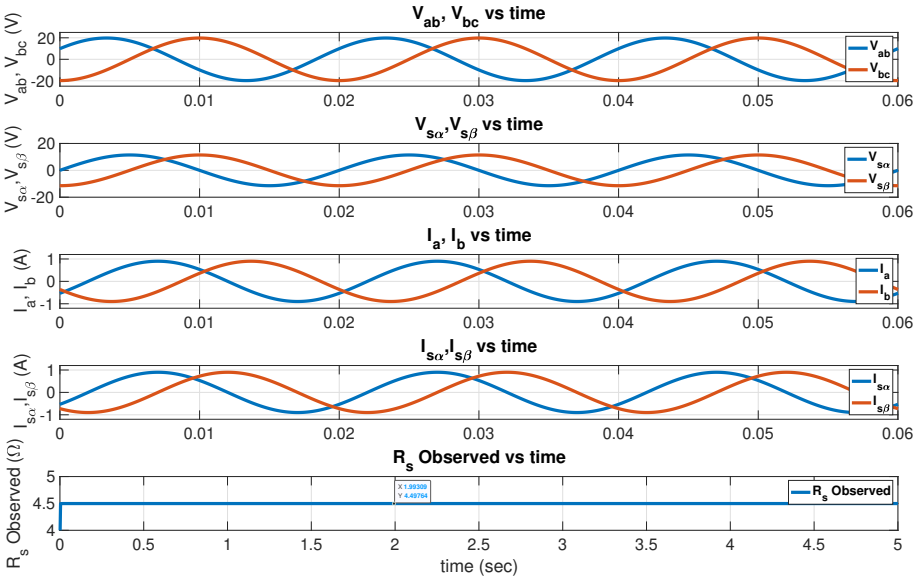


Figure 4.5: Standstill simulation result for test motor without any series resistance

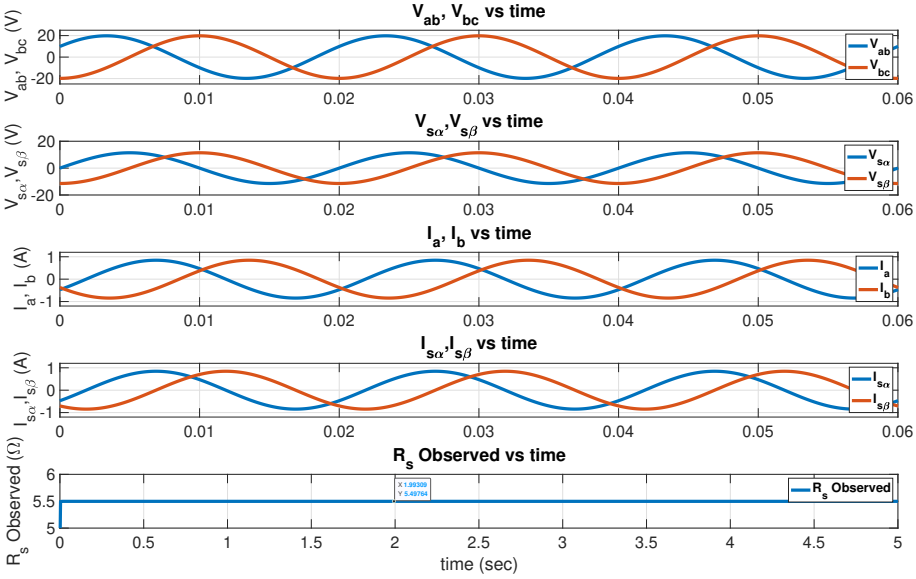


Figure 4.6: Standstill simulation result for test motor with 1 Ohm series resistance

4.2.2 Stator Resistance Estimation for Running Condition

This simulation is done while the motor is running at 3000 RPM. The window length is chosen as N=4 and the diagonal elements of noise covariance matrix $R_k[A]$ value is chosen as $\sigma = 0.000459$. Figure 4.7 shows the simulation result for the stator resistance $R_s = 4.45\,\Omega$. Figure 4.8 shows the simulation result for stator resistance with an additional series $1\,\Omega$ resistance.

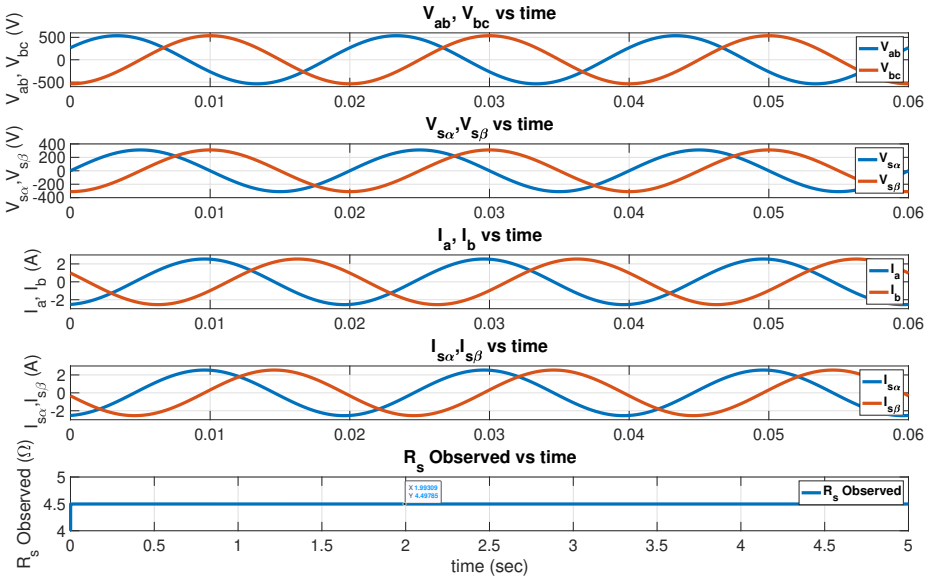


Figure 4.7: Variables used to estimate $\mathbf{R_s}$ at 3000 RPM without 1 Ohm series resistance

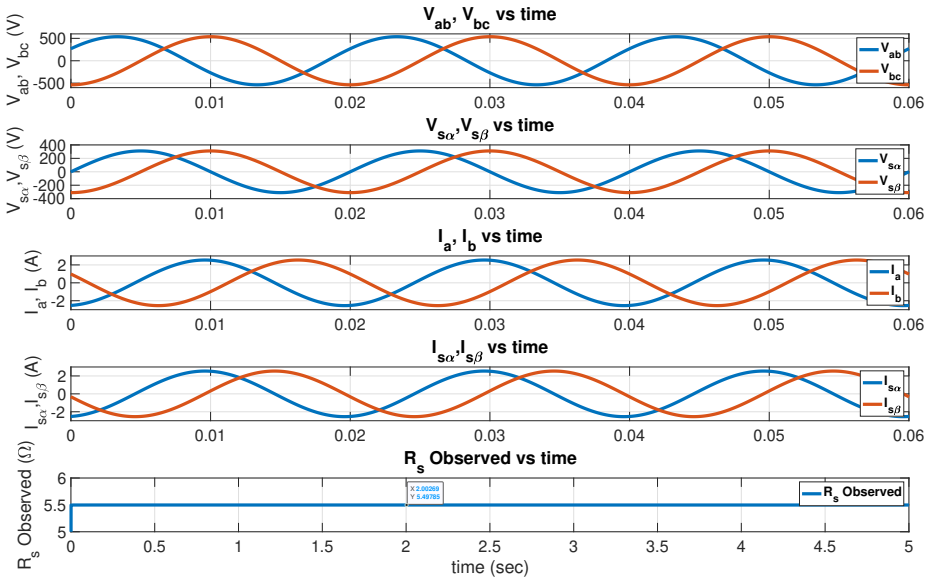


Figure 4.8: Variables used to estimate $\mathbf{R_s}$ at 3000 RPM with 1 Ohm series resistance

4.3 Conclusion

From the simulation results, we can conclude the IAEKF algorithm can detect the stator resistance value and any change in stator resistance value successfully in the simulation environment at standstill and running conditions. Extensive simulations are done in the simulation environment to see the effect off various window lengths, various noise covariance matrix values, and various frequencies. After verifying the model in the simulation environment, we were ready to implement this algorithm in DSPACE DS1104, which operates in the real-time. The following chapters investigates the IAEKF algorithm's performance in the real-time environment.

CHAPTER 5

EXPERIMENT RESULTS FOR STANDSTILL TESTS

5.1 Introduction

In order to verify the IAEKF algorithm's performance in the real-time environment, the parameters of the motor must be determined accurately. The algorithm directly utilizes parameters such as rotor resistance, magnetizing inductance and leakage inductance while it estimates the stator resistance. To determine the parameters of the motor, no-load and locked rotor tests are done according to the test procedure described in [11]. The stator resistance is measured by a high-precision four-wire resistance meter and is found to be 4.501 Ω. The rotor resistance, magnetizing inductance and leakage inductance are determined according to the procedure recommended by IEEE. Voltage and current sensors with high accuracy must be utilized to measure the three-phase voltages and currents of the induction motor. In the standstill tests, the motor does not have to be connected to any load and is not required to have any speed sensor. However, in the following chapter, the performance of the algorithm is also tested under loaded conditions and at running tests. An encoder is mounted on the shaft of the test motor to make sure the IAEKF algorithm is not affected incase the sensorless speed estimation has some error. In the IAEKF algorithm, encoder output is used as speed input. Note that the speed information is zero at the standstill tests. The test setup is given in Figure 5.1.

The estimation performance of EKF is sensitive to the execution time and can be improved by reducing the sampling time as mentioned in [46], [47], [48], [49], [50]. In addition to these papers, the effect of sampling time on the stability of EKF is examined in detail in [47]. It can be understood that by reducing the execution time,

the stochastic-based estimator's estimation performances can be improved since the discrete models of the systems approach their continuous-time models with high accuracy. Our drive system's sampling time is determined as 200 µs. This sampling time is the minimum execution time that the ControlDesk interface allows and that can implement the control and estimation algorithms with measurements on DS1104 DSP. The results are recorded by the ControlDesk interface and plotted with Matlab.

In the experiments, the induction motor is supplied from a 3-phase variable voltage transformer and V/f controlled PWM inverter. The reason for testing the algorithm with a 3-phase variable voltage transformer was to see whether the PWM inverter has any effect on the estimation performance of the IAEKF algorithm. In the standstill tests the voltage applied to the motor is kept small so that the motor is unable to spin. In the experiments the window length N value and the diagonal elements of the noise covariance matrix $R_k[A]$ are changed to observe their effect on the estimation performance of IAEKF algorithms estimation performance.

5.1.1 Experimental Procedure

The algorithm is tested firstly with a motor supplied by a variable voltage transformer; the voltage to be applied is set by a 3-phase variable voltage transformer. The phase voltage of the motor has a magnitude of 19.5 V_{peak} and a frequency of 50Hz. The experiments were done for different N values (4, 8, 16, 32, 64) and different $R_k[A]$ values (0.000230, 0.000275, 0.000321, 0.000367, 0.000413, 0.000459, 0.000505, 0.000551, 0.000597, 0.000643, 0.000689). All of the experiments are recorded for 5 seconds and their mean values are calculated. The variable voltage transformer experiments were done to eliminate the PWM inverter's nonlinear behavior and verify the success of the algorithm in the real-time environment. The same tests are repeated with 1 Ω series resistor added to the stator windings to verify the algorithm can detect the changes in the stator resistance value.

After the verification of the algorithm with the 3-phase variable voltage transformer, experiments with V/f controlled PWM inverter are conducted for different frequency values (10 Hz, 25 Hz, 50 Hz, 75 Hz, 100 Hz), different N values (4, 8, 16, 32, 64) and different $R_k[A]$ values (0.000230, 0.000275, 0.000321, 0.000367, 0.000413,

0.000459, 0.000505, 0.000551, 0.000597, 0.000643, 0.000689). The phase voltage of the motor has a magnitude of 19.5 V_{peak} for all frequencies. All tests described above are repeated with a 1 Ω series resistor added to the stator windings to verify that the algorithm can detect any change in the stator resistance value. It has been observed that the amplitude of the applied voltage has no effect on the stator resistance estimation result of the algorithm, which can be seen in Figure 5.4. The block diagram of the tests conducted by the PWM inverter is given in the Figure 5.1. The starting value for the R_s is chosen as 4.45 Ω, to converge faster to the result, however, the initial value does not have any effect on the estimation result after a long period of time, we have shown the convergence performance of the estimator for different initial R_s condition in Figure 5.5 and in Figure 5.6.

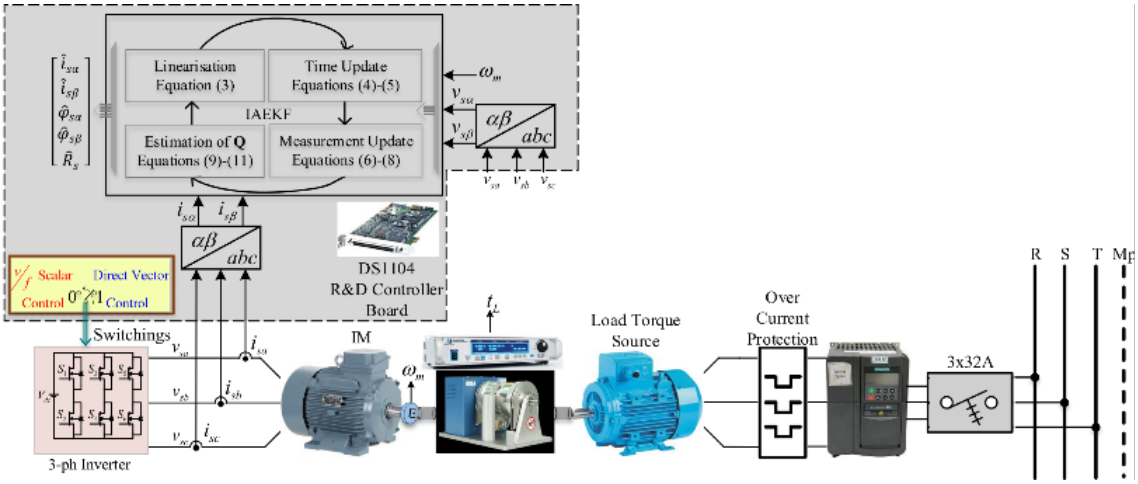


Figure 5.1: Block diagram of the experimental setup conducted with an Inverter

We have conducted extensive experiments, to make these experiments meaningful we have prepared tables from the experiment results. Table 5.1 shows the result of $\hat{R}_s$ estimation with variable voltage transformer and inverter for different frequencies. The results show that the skin effect and proximity effect can be detected by the IAEKF algorithm successfully. The error is calculated by "% error=100x($\hat{R}_s$-R_s)/R_s" equation in the following tables, where R_s value is 4.5 Ω.

Table 5.2 shows the result of $\hat{R}_s$ estimation with variable voltage transformer and inverter for different frequencies when a series 1 Ω series resistance is added. The results show that the skin effect and proximity effect can be detected by the IAEKF algorithm successfully. In addition, the change in the resistance value can also be

Table 5.1: Estimated $\hat{R}_s$ and percentage error for variable voltage transformer and PWM supply with different frequencies are tested, window length N=4 and $R_k[A^2] = 0.000459$

Voltage Supply	$\hat{R}_s(\Omega)$	% Error
Variable voltage Transformer	4.6332	+2.9
10 Hz PWM Supply	4.3591	-3.1
25 Hz PWM Supply	4.4180	-1.8
50 Hz PWM Supply	4.6417	+3.1
75 Hz PWM Supply	4.7380	+5.2
100 Hz PWM Supply	4.8533	+7.8

Table 5.2: Estimated $\hat{R}_s$ and percentage error for variable voltage transformer and PWM supply with different frequencies are tested while series 1 Ω resistance is added, window length N=4 and $R_k[A^2] = 0.000459$

Voltage Supply	$\hat{R}_s(\Omega)$+1 (Ω)	% Error
Variable Voltage Transformer	5.5941	+1.69
10 Hz PWM Supply	5.3071	-3.52
25 Hz PWM Supply	5.4368	-1.17
50 Hz PWM Supply	5.6590	+2.87
75 Hz PWM Supply	5.7364	+4.28
100 Hz PWM Supply	5.8418	+6.2

Table 5.3: Estimated $\hat{R}_s$ value for various window length N values at 50Hz while $R_k[A^2] = 0.000459$

N	4	8	16	32	64
$\hat{R}_s(\Omega)$	4.6428	4.6532	4.6481	4.6326	4.6490

Table 5.4: Estimated $\hat{R}_s$ value for various $R_k[A]$ values at 50 Hz while the window length is N=4

$R_k[A]$	0.000230	0.000321	0.000413	0.000459	0.000505	0.000597	0.000689
$\hat{R}_s(\Omega)$	4.6330	4.6338	4.6391	4.6428	4.6504	4.6529	4.6558

detected by this algorithm. This property makes this algorithm suitable for R_s estimation for self-commissioning purposes.

Table 5.3 shows the estimation of the $\hat{R}_s$ value for different window length N values. It can be concluded that even a window length of 4 is enough to estimate the $\hat{R}_s$ value accurately.

Table 5.4 shows the estimation of the $\hat{R}_s$ value for different $R_k[A]$ values. Calculated $R_k[A]$ values appear to work insensitive to $R_k[A]$ in quite a wide range around the calculated value.

The IAEKF algorithm is used to estimate the stator resistance for a given range of diagonal element values R_k and N when the PWM inverter is driven at 50Hz. Figure 5.2 depicts the graphical results of the tests performed for the test motor. The tests are carried out again with an external 1 Ω series resistance are shown in Figure 5.3.

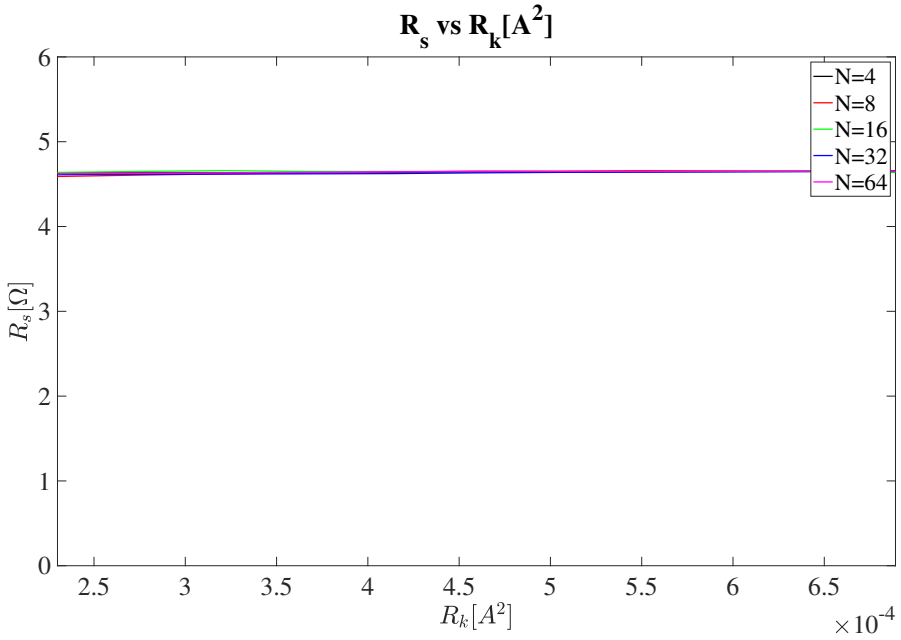


Figure 5.2: Measurements done with PWM inverter at 50 Hz for different N and $R_k[A^2]$ values

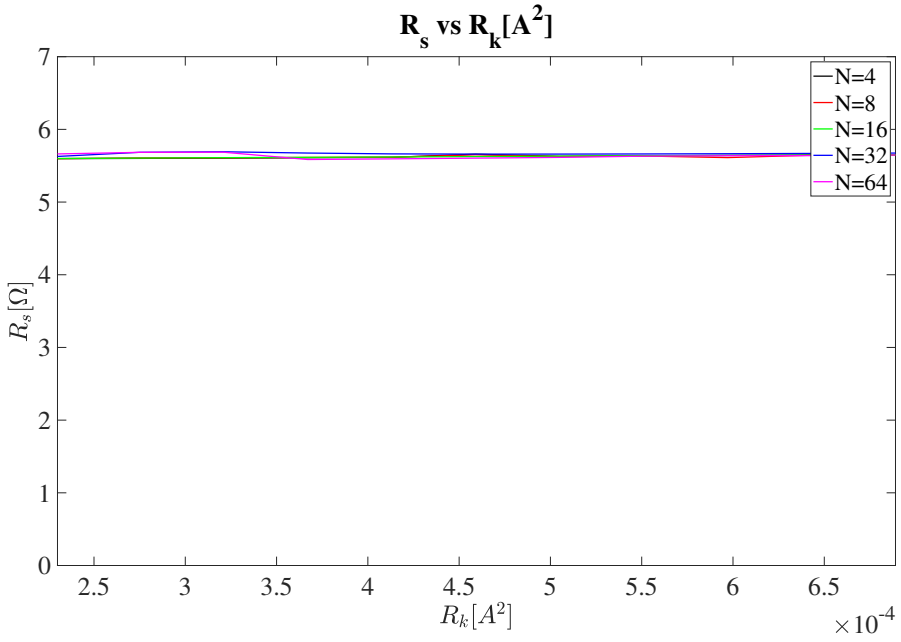


Figure 5.3: Measurements done with PWM inverter at 50 Hz for different N and $R_k[A^2]$ values, 1 Ω series resistance added

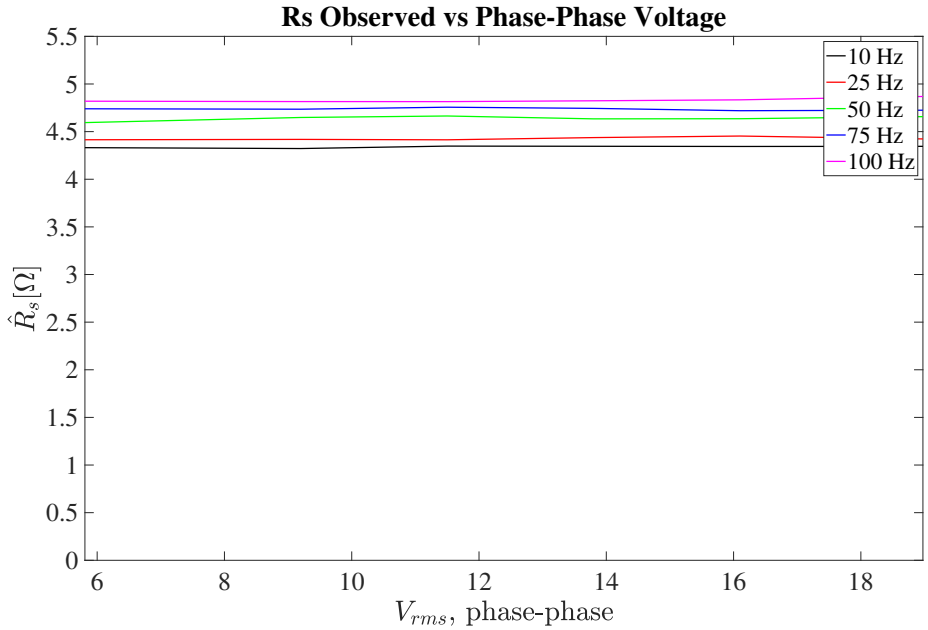


Figure 5.4: R_s for different V_{rms} and different frequencies

Figure 5.4 shows the amplitude of the applied voltage has no effect on the stator resistance estimation in the given interval. However, the frequency has an influence on the stator resistance estimation due to the skin and proximity effect.

Figure 5.5 shows stator resistance estimation for a starting value of $4.45\ \Omega$. In our experiments, we preferred to assign initial value of $4.45\ \Omega$ for the $\hat{R}_s$ value. By doing so we were able to converge faster to the estimated value, however giving zero initial value for $\hat{R}_s$ is also possible, only the convergence time is longer for this case. Figure 5.6 shows the convergence of the $\hat{R}_s$ value to the estimated value for zero initial condition. However the experiment results given for 5 seconds do not include these transients. Our aim to show $\hat{R}_s$ result for 5 seconds was to illustrate the estimation results are stable and accurate for a long period of time since the thermal time constant of the motor is much larger than the electrical time constant. These experiments are conducted at 50Hz with a window length as N=4 and diagonal elements of the noise covariance matrix as $R_k[A^2] = 0.000459$.

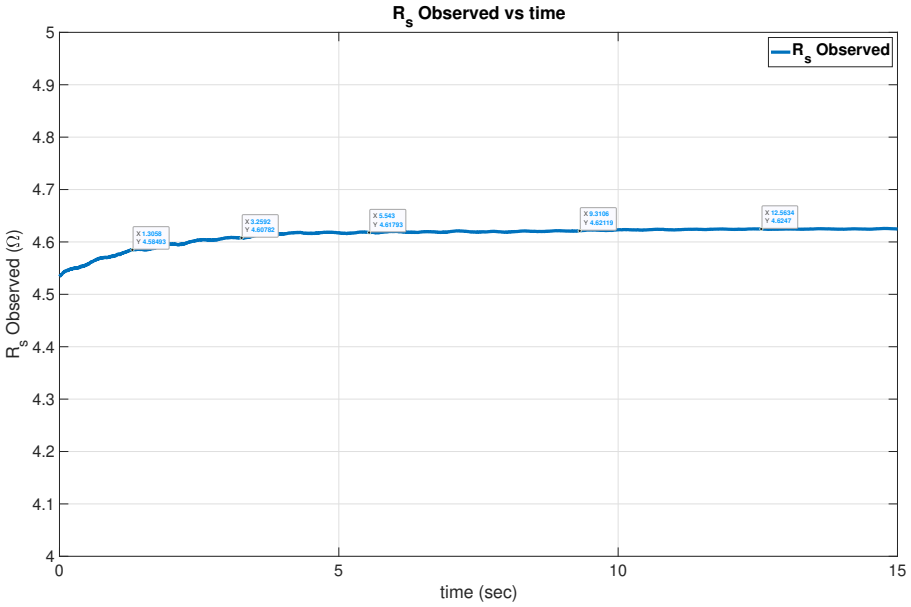


Figure 5.5: Convergence of the estimator when the initial condition of $\hat{R}_s$ is $4.45\,\Omega$

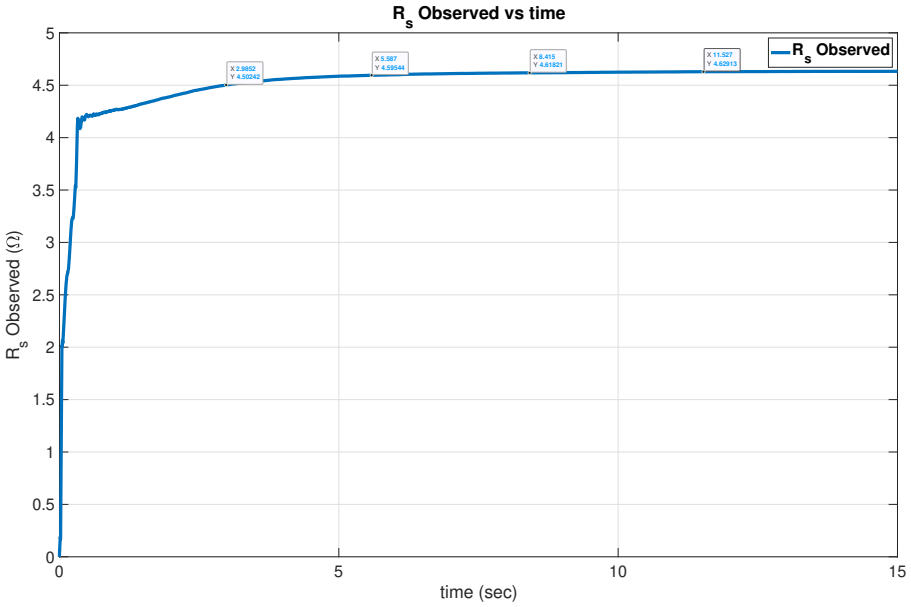


Figure 5.6: Convergence of the estimator when the initial condition of $\hat{R}_s$ is $0\,\Omega$

5.1.1.1 Standstill Test Conducted with Variable Voltage Transformer

Standstill estimation performance of the IAEKF algorithm is tested while the induction motor is driven with a 3-phase variable voltage transformer at 50 Hz. The estimation result on R_s is given in Figure 5.7. Then an additional 1 Ω resistance is added to the stator winding of the motor to verify that IAEKF algorithm can detect any change in the resistance value, the experimental plot of this experiment is given in Figure 5.8. The following experimental plots are the results of experiments conducted with window length as N=4 and diagonal elements of the noise covariance matrix as $R_k[A^2] = 0.000459$.

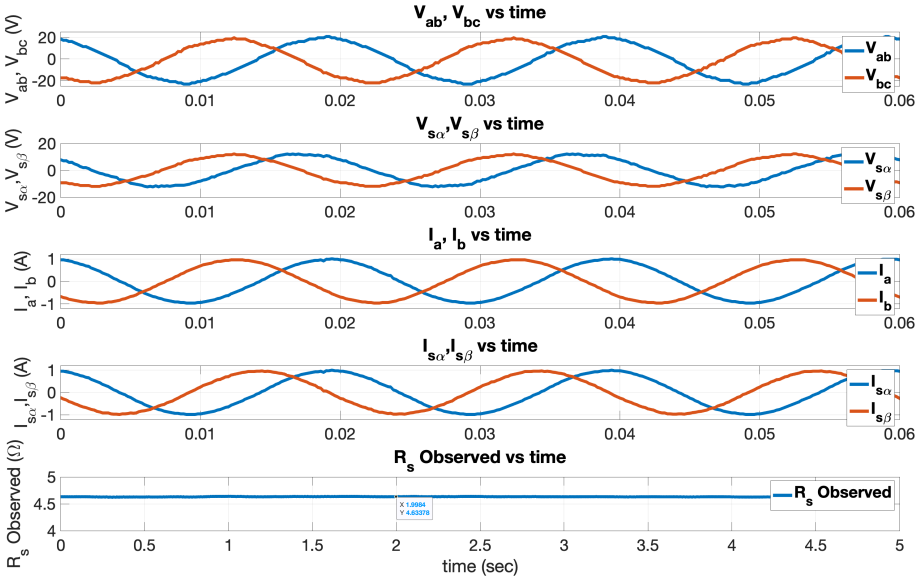


Figure 5.7: Variables used to estimate $\mathbf{R_s}$ at 50 Hz without 1 Ω series resistance

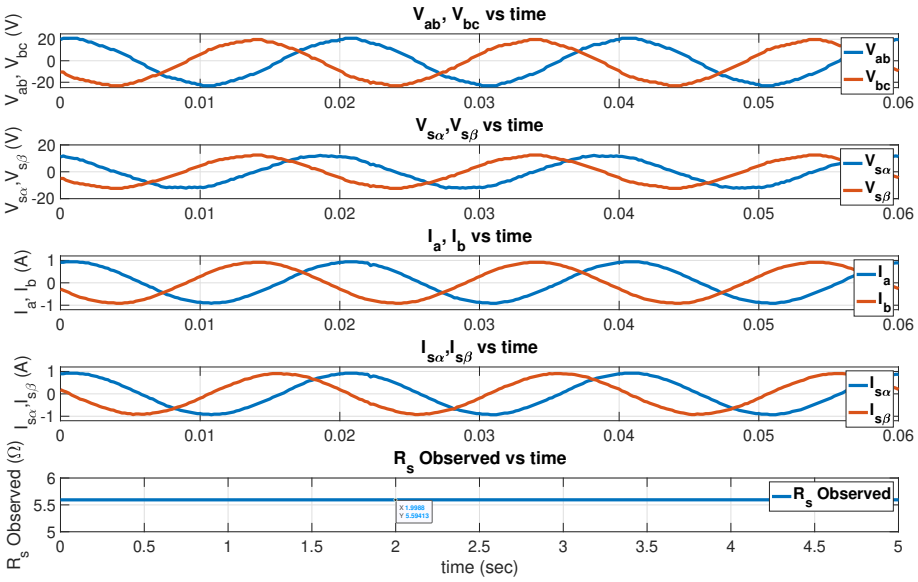


Figure 5.8: Variables used to estimate $\mathbf{R_s}$ at 50 Hz with 1 Ω series resistance

5.1.1.2 Standstill Test Conducted with PWM Inverter Driven at 10Hz

Standstill estimation performance of the IAEKF algorithm is tested while the induction motor is driven with a PWM inverter at 10 Hz. The estimation result on $\mathbf{R_s}$ is given in Figure 5.9. Then an additional 1 Ω resistance is added to the stator winding of the motor to verify that IAEKF algorithm can detect any change in the resistance value, the experimental plot of this experiment is given in Figure 5.10. The following experimental plots are the results of experiments conducted with window length as N=4 and diagonal elements of the noise covariance matrix as $R_k[A^2] = 0.000459$.

From the results, it can be concluded that the skin effect and the proximity effect are not dominant at low frequencies. Figure 5.10 shows that the change in stator resistance value can be detected successfully by the IAEKF algorithm. The stator resistance change detection performance is very crucial for the vector control algorithm.

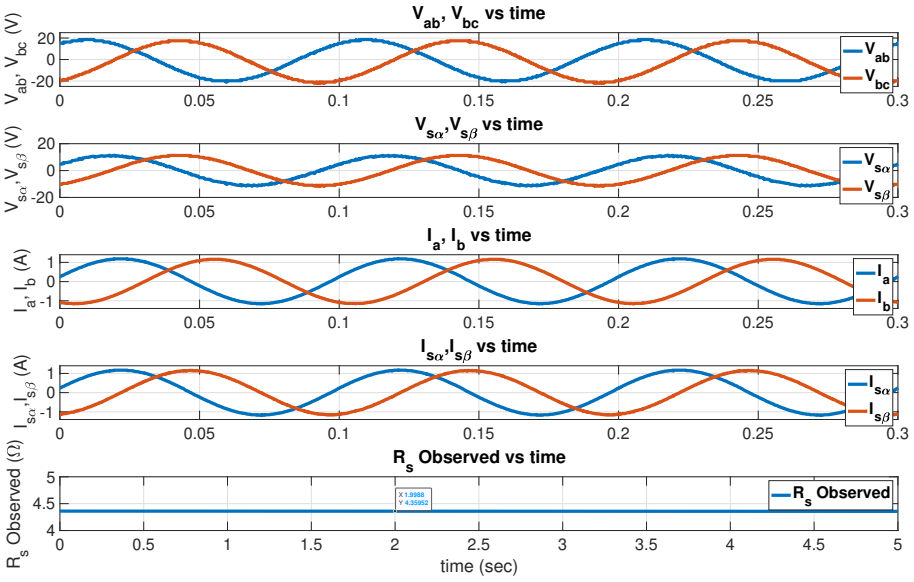


Figure 5.9: Variables used to estimate $\mathbf{R_s}$ at 10 Hz without 1 Ω series resistance

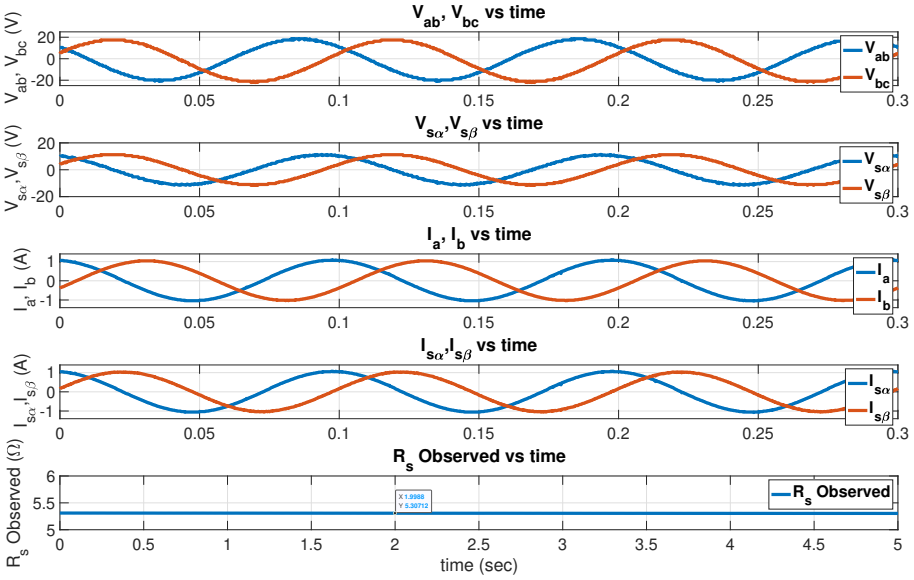


Figure 5.10: Variables used to estimate $\mathbf{R_s}$ at 10 Hz with 1 Ω series resistance

5.1.1.3 Standstill Test Conducted with PWM Inverter Driven at 25Hz

Standstill estimation performance of the IAEKF algorithm is tested while the induction motor is driven with a PWM inverter at 25 Hz. The estimation result on $\mathbf{R_s}$ is given in Figure 5.11. Then an additional 1 Ω resistance is added to the stator winding of the motor to verify that IAEKF algorithm can detect any change in the resistance value, the experimental plot of this experiment is given in Figure 5.12. The following experimental plots are the results of experiments conducted with window length as N=4 and diagonal elements of the noise covariance matrix as $R_k[A^2] = 0.000459$.

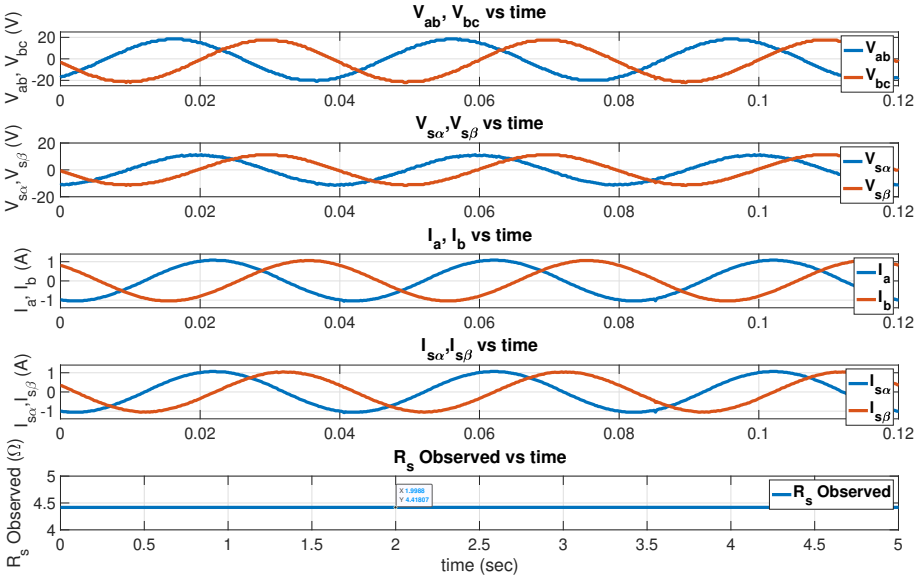


Figure 5.11: Variables used to estimate $\mathbf{R_s}$ at 25 Hz without 1 Ω series resistance

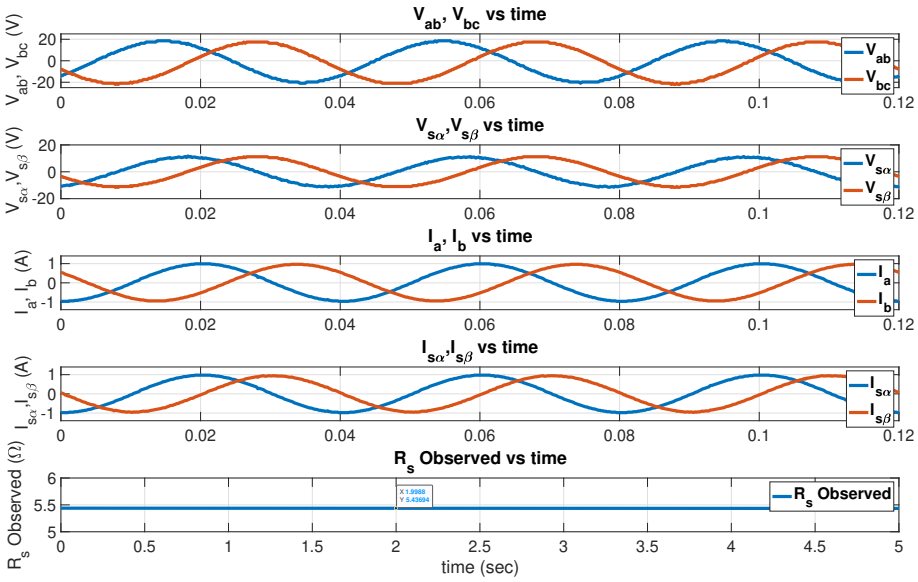


Figure 5.12: Variables used to estimate $\mathbf{R_s}$ at 25 Hz with 1 Ω series resistance

5.1.1.4 Standstill Test Conducted with PWM Inverter Driven at 50Hz

Standstill estimation performance of the IAEKF algorithm is tested while the induction motor is driven with a PWM inverter at 50 Hz. The estimation result on $\mathbf{R_s}$ is given in Figure 5.13. Then an additional 1 Ω resistance is added to the stator winding of the motor to verify that IAEKF algorithm can detect any change in the resistance value, the experimental plot of this experiment is given in Figure 5.14. The following experimental plots are the results of experiments conducted with window length as N=4 and diagonal elements of the noise covariance matrix as $R_k[A^2] = 0.000459$.

Figure 5.14 shows that the change in stator resistance value can be detected successfully by the IAEKF algorithm.

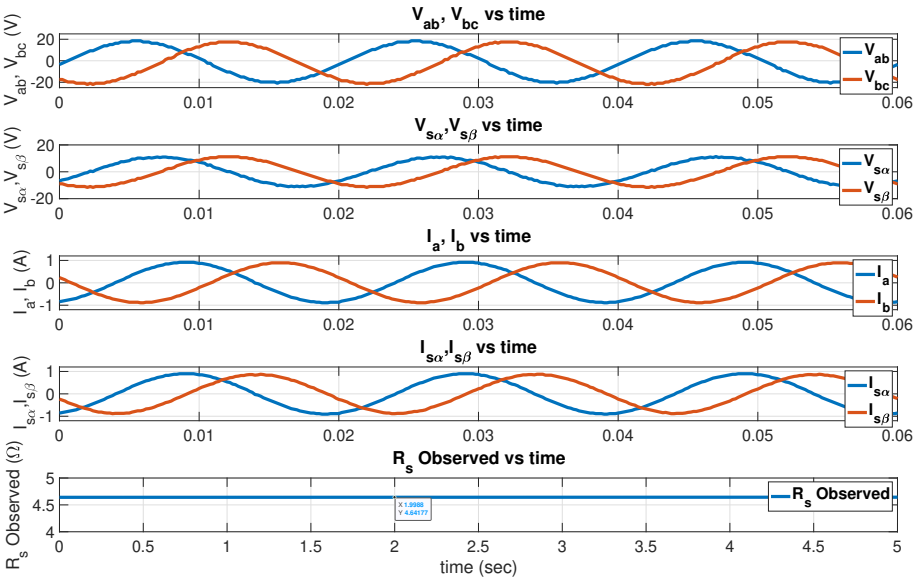


Figure 5.13: Variables used to estimate $\mathbf{R_s}$ at 50 Hz without 1 Ω series resistance

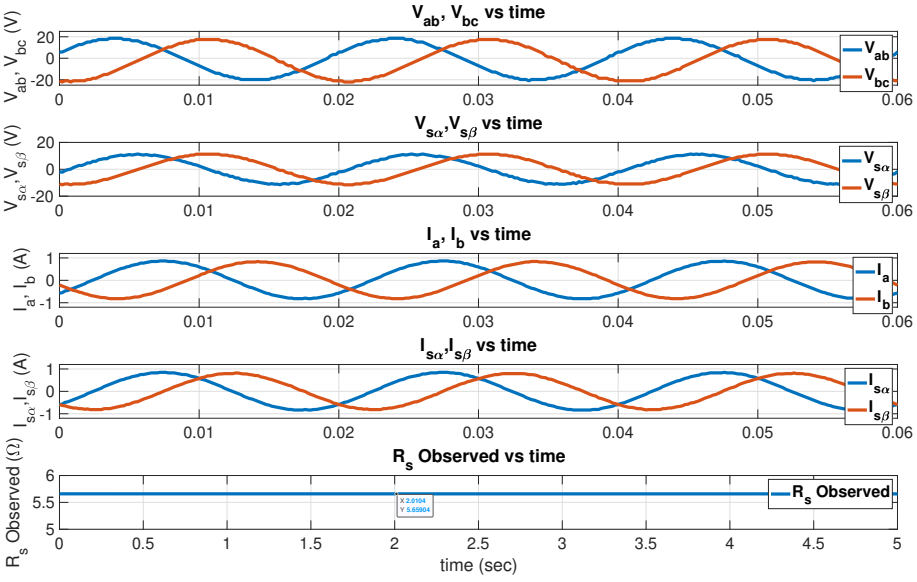


Figure 5.14: Variables used to estimate $\mathbf{R_s}$ at 50 Hz with 1 Ω series resistance

5.1.1.5 Standstill Test Conducted with PWM Inverter Driven at 75Hz

Standstill estimation performance of the IAEKF algorithm is tested while the induction motor is driven with a PWM inverter at 75 Hz. The estimation result on $\mathbf{R_s}$ is

given in Figure 5.15. Then an additional 1 Ω resistance is added to the stator winding of the motor to verify that IAEKF algorithm can detect any change in the resistance value, the experimental plot of this experiment is given in Figure 5.15. The following experimental plots are the results of experiments conducted with window length as N=4 and diagonal elements of the noise covariance matrix as $R_k[A^2] = 0.000459$.

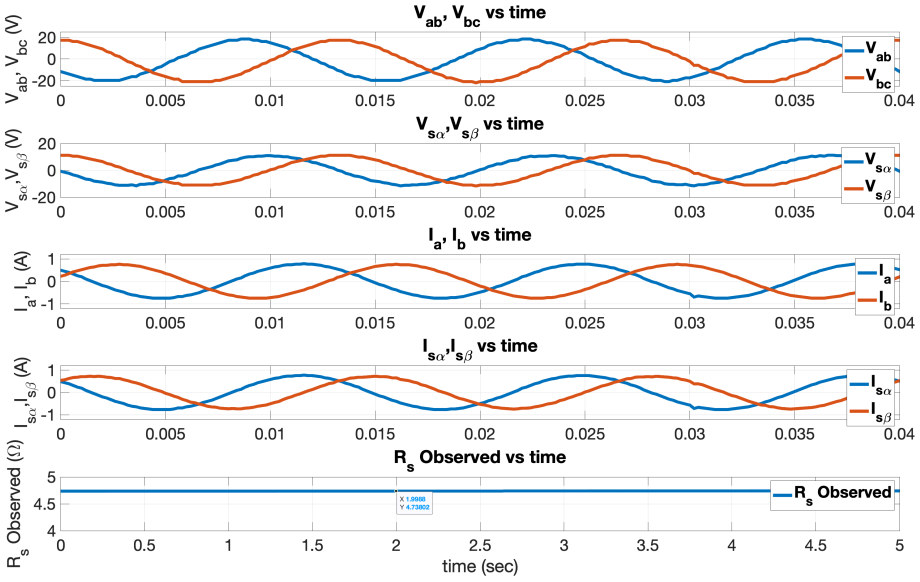


Figure 5.15: Variables used to estimate $\mathbf{R_s}$ at 75 Hz without 1 Ω series resistance

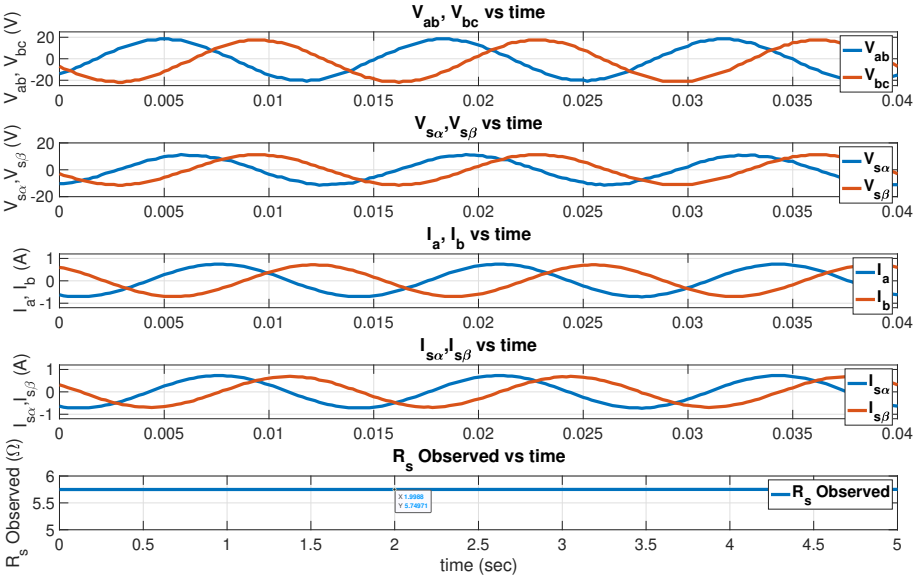


Figure 5.16: Variables used to estimate $\mathbf{R_s}$ at 75 Hz with 1 Ω series resistance

5.1.1.6 Standstill Test Conducted with PWM Inverter Driven at 100Hz

Standstill estimation performance of the IAEKF algorithm is tested while the induction motor is driven with a PWM inverter at 100 Hz. The estimation result on $\mathbf{R_s}$ is given in Figure 5.17. Then an additional 1 Ω resistance is added to the stator winding of the motor to verify that IAEKF algorithm can detect any change in the resistance value, the experimental plot of this experiment is given in Figure 5.18. The following experimental plots are the results of experiments conducted with window length as N=4 and diagonal elements of the noise covariance matrix as $R_k[A^2] = 0.000459$.

From the results, it can be concluded that the skin effect and proximity effect are dominant at high frequencies. From all of these tests, we can conclude that IAEKF estimator accounts for the skin and proximity effect on stator resistance, resulting in a more realistic calculation of stator resistance.

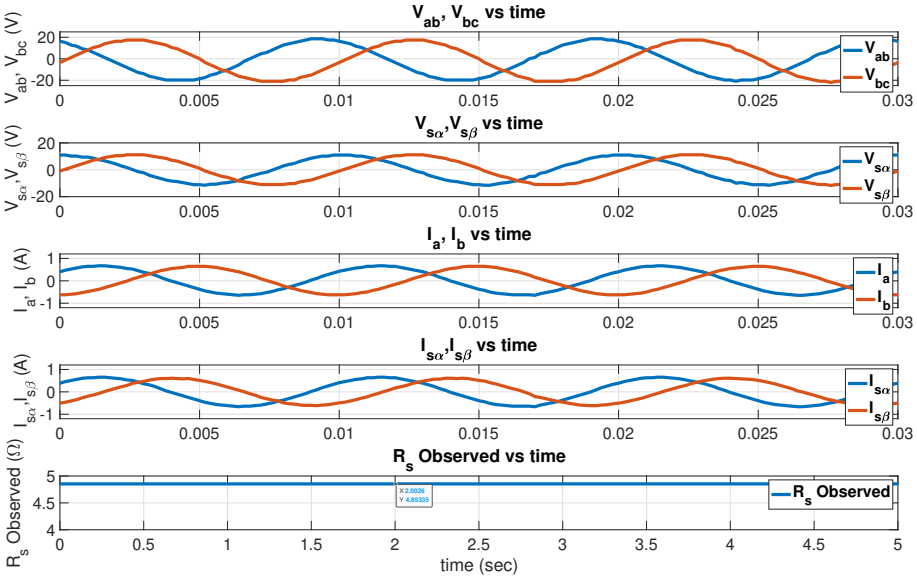


Figure 5.17: Variables used to estimate $\mathbf{R_s}$ at 100 Hz without 1 Ω series resistance

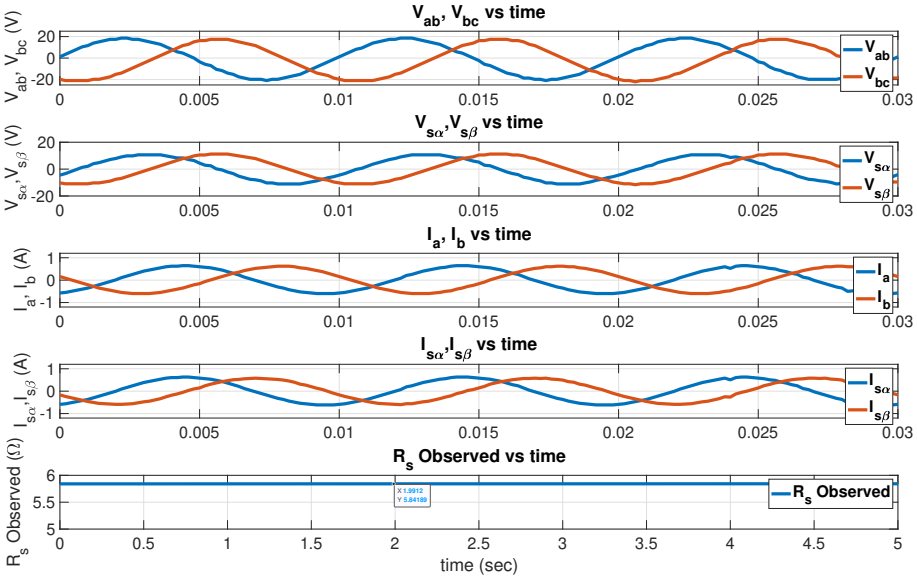


Figure 5.18: Variables used to estimate $\mathbf{R_s}$ at 100 Hz with 1 Ω series resistance

Figure 5.19 shows in addition to R_s value estimation, $\alpha\beta$ components of stator current $i_{s\alpha}$, $i_{s\beta}$ and stator flux $\varphi_{s\alpha}$, $\varphi_{s\beta}$ can also be estimated. It can be seen that measured and estimated values of stator currents ($i_{s\alpha}$, $\hat{i}_{s\alpha}$, $i_{s\beta}$, $\hat{i}_{s\beta}$) completely overlap. The stator flux is also predicted and the amplitude of the flux is low since the applied

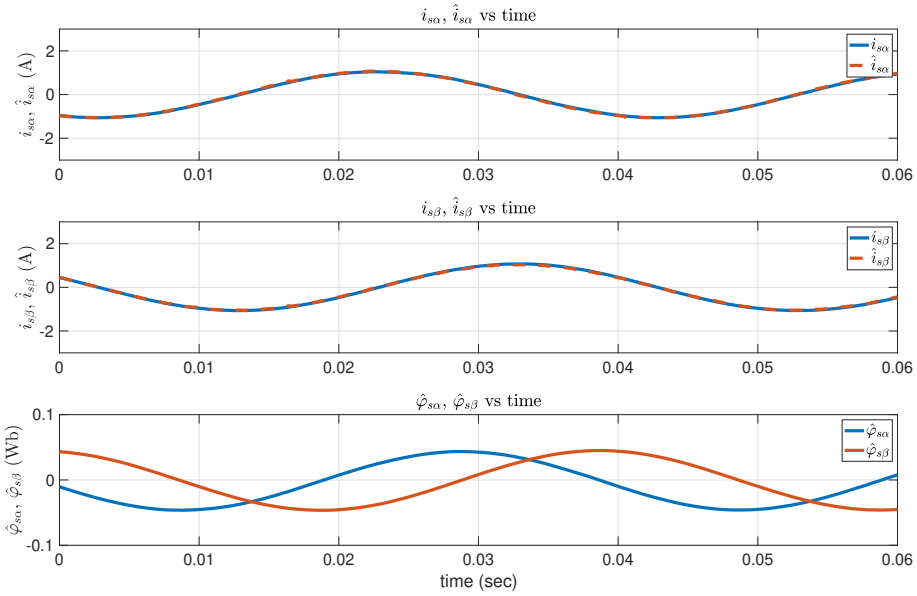


Figure 5.19: Estimation of $\alpha\beta$ components of stator current $i_{s\alpha}$, $i_{s\beta}$ and stator flux $\varphi_{s\alpha}$, $\varphi_{s\beta}$

voltage is also very small at the standstill tests. The phase voltage of the motor has a magnitude of 19.5 V_{peak} and frequency of 25Hz in this experiment. The IAEKF has window length of N=4 and $R_k[A^2]$=0.000459.

5.2 Conclusion

In this chapter, it is shown that a novel IAEKF estimator may be used to estimate the induction motor's stator resistance R_s at standstill before an inverter-driven IM starts operating. Both with a sinusoidal power supply and with a PWM voltage supply, the proposed IAEKF's estimate performance is tested in two separate open-loop real-time induction motor driver experimental setups. The experiments are done firstly by a variable voltage transformer as a power supply. Lastly, the experiments are done with PWM power supply to test the performance of the IAEKF estimator performance at different frequencies. Finally, these tests reveal that the proposed IAEKF has an excellent standstill performance, particularly on determining the stator resistance value R_s.

CHAPTER 6

EXPERIMENT RESULTS FOR RUNNING TESTS

6.1 Introduction

In this section, online stator resistance estimation is done with the novel IAEKF algorithm while the motor is running with sensorless vector control with the speed loop activated. In these experiments, the purpose is to test the IAEKF algorithm's performance for different diagonal elements of R_k matrix and different frequencies while the motor is running. After these experiments, an additional 1 Ω resistance is added to the stator windings to observe the performance of IAEKF algorithm in the conditions where the stator resistance value changes. The motor is driven while there is a load motor connected as seen in Figure 6.8, however in the following test the load motor was not activated. The motor resistance is measured just before and after the experiments. The tests are kept short so that the stator resistance value of the motor is not permitted to rise. We kept the time between the experiments long to cool the motor, we did not start the following experiments before the stator resistance becomes 4.5 Ω after each experiment. Lastly, we have tested the performance of the estimator under load torque disturbances with a value of rated torque in Section 6.2.5.

6.2 Experimental Procedure

The IAEKF algorithm is tested with two-level vector-controlled PWM inverter. The experiments are conducted for different frequencies (10 Hz, 25 Hz, 50 Hz), window length of N=4 value, and different diagonal elements of R_k matrix (0.5, 1, 1.5). The $R_k[A]$ values in the standstill experiments did not work at running experiments, we

used the trial and error method to adjust $R_k[A]$ values. All of the experiments are recorded for 5 seconds, and their mean values are calculated. All of the tests described above are repeated with a 1 Ω series resistor added to the stator windings to verify that the algorithm can detect successfully the changes in the stator resistance value. The experimental setup can be seen in Figure 6.8. Contrary to the results in the simulation environment we can observe that the estimation result changes with the speed. This mismatch is caused by the skin and the proximity effect.

Table 6.1: Estimated $\hat{R}_s$ value at different speeds, window length N=4 and $R_k[A^2] = 1$

Speed	$\hat{R}_s$ (Ω)	% Error	$\hat{R}_s$ (Ω)+1 (Ω)	% Error
600 RPM	4.3759	-2.78	5.4099	-1.66
1500 RPM	4.4604	-0.9	5.4962	-0.01
3000 RPM	4.6026	+2.26	5.5633	+1.13

Table 6.1 shows the result of $\hat{R}_s$ estimation results of the IAEKF algorithm with vector-controlled inverter operated at 600, 1500, and 3000 RPM, and the window length N=4 and $R_k[A^2] = 1$. We see that depending upon the speed at which measurements are made, the predicted $\hat{R}_s$ values display about %3 variation from the DC resistance measured value. Because of this reason, the effect of this range of resistance prediction error and predicted motor speed is investigated in the following section. The error is calculated by "% error=100x($\hat{R}_s$-R_s)/R_s" equation in the following tables, where R_s value is 4.5 Ω.

6.2.1 $\mathbf{R_s}$ Estimation at 600 RPM

Running estimation performance of the IAEKF algorithm is tested while the induction motor is driven with PWM supply and the motor is running at 600 RPM speed. The estimation result on $\mathbf{R_s}$ is given in Figure 6.1 and $\mathbf{R_s}$ with 1 Ω series resistance added is given in Figure 6.2. The following experimental plots are results of experiments conducted with window length as N=4 and diagonal elements of the noise covariance as $R_k[A^2] = 1$.

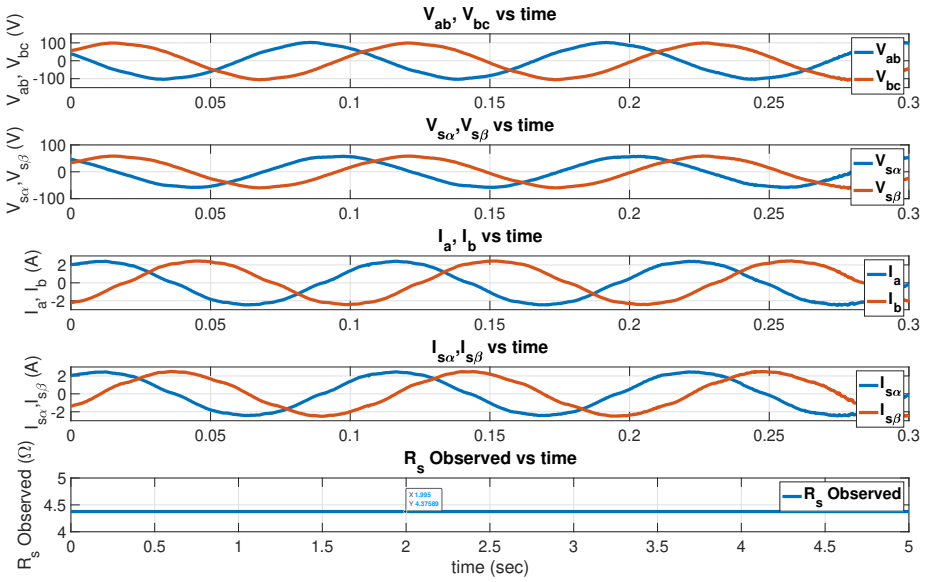


Figure 6.1: Variables used to estimate $\mathbf{R_s}$ at 600 RPM without 1 Ω series resistance

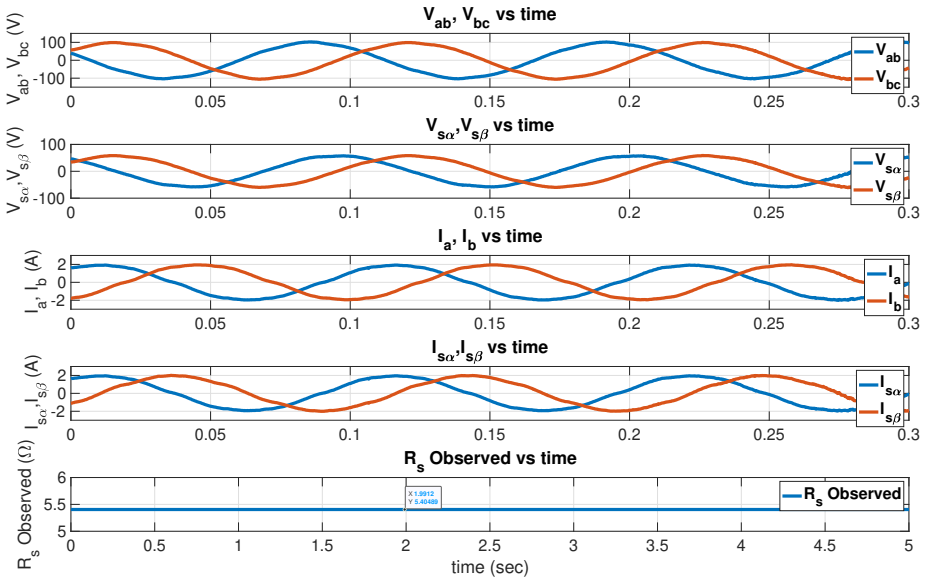


Figure 6.2: Variables used to estimate $\mathbf{R_s}$ at 600 RPM with 1 Ω series resistance

6.2.2 $\mathbf{R_s}$ Estimation at 1500 RPM

Running estimation performance of the novel IAEKF is tested while the induction motor is driven with PWM inverter at 1500 RPM. The estimation result on $\mathbf{R_s}$ is

given in Figure 6.3 and $\mathbf{R_s}$ with 1 Ω series resistance added is given in Figure 6.4. The following experimental plots are results of experiments conducted with window length as N=4 and diagonal elements of the noise covariance as $R_k[A^2] = 1$.

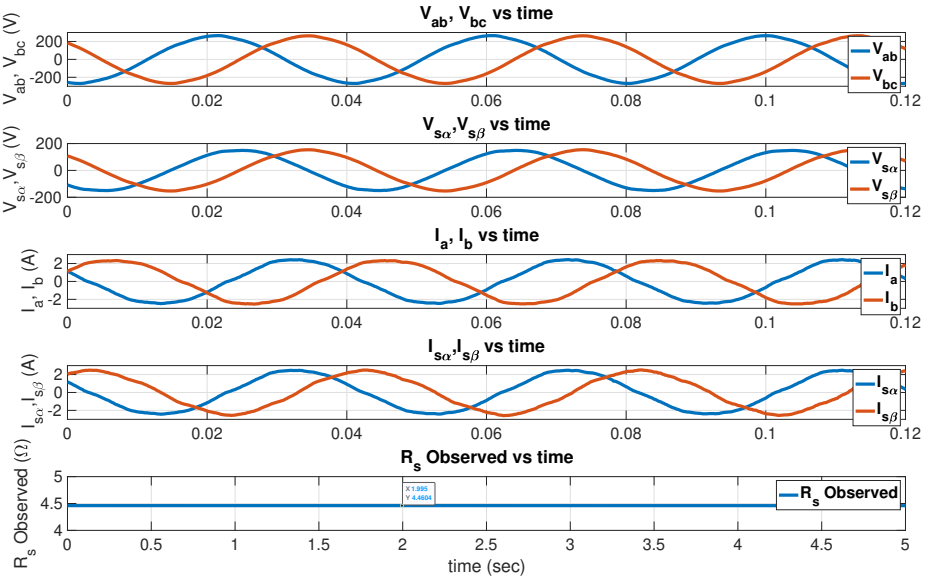


Figure 6.3: Variables used to estimate $\mathbf{R_s}$ at 1500 RPM without 1 Ω series resistance

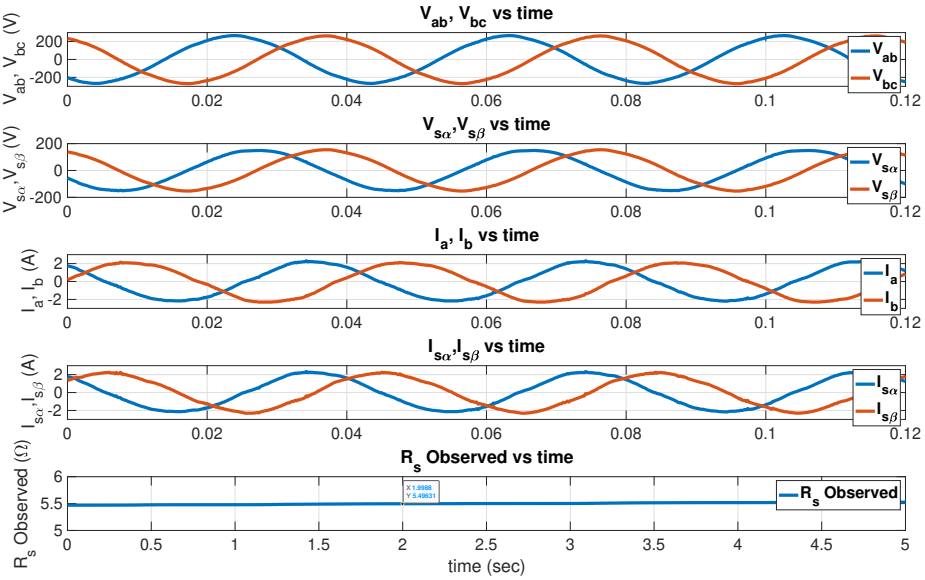


Figure 6.4: Variables used to estimate $\mathbf{R_s}$ at 1500 RPM with 1 Ω series resistance

6.2.3 $\mathbf{R_s}$ Estimation at 3000 RPM

Running estimation performance of the IAEKF algorithm is tested while the induction motor is driven with PWM inverter at 3000 RPM. The estimation result on $\mathbf{R_s}$ is given in Figure 6.3 and $\mathbf{R_s}$ with 1 Ω series resistance added is given in Figure 6.4. The following experimental plots are results of experiments conducted with window length as N=4 and diagonal elements of the noise covariance as $R_k[A^2] = 1$.

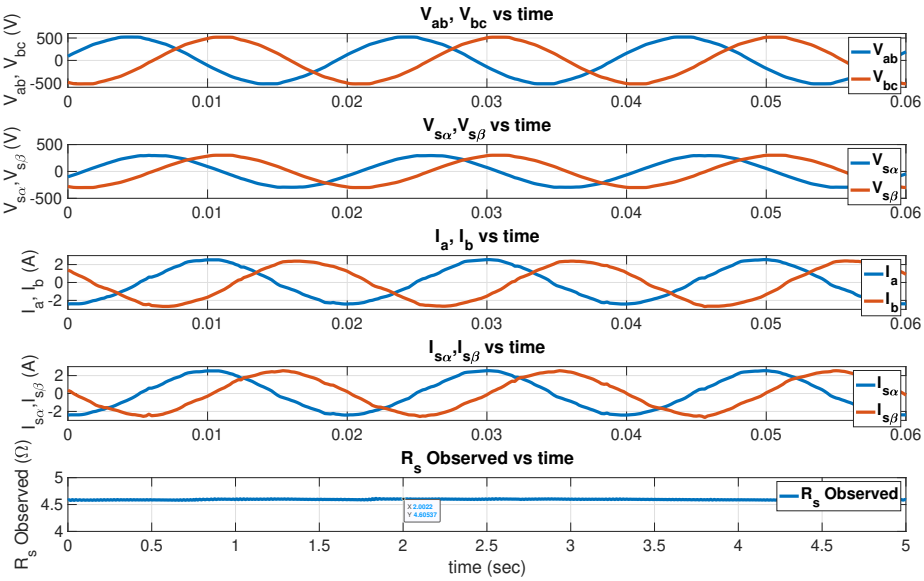


Figure 6.5: Variables used to estimate $\mathbf{R_s}$ at 3000 RPM without 1 Ω series resistance

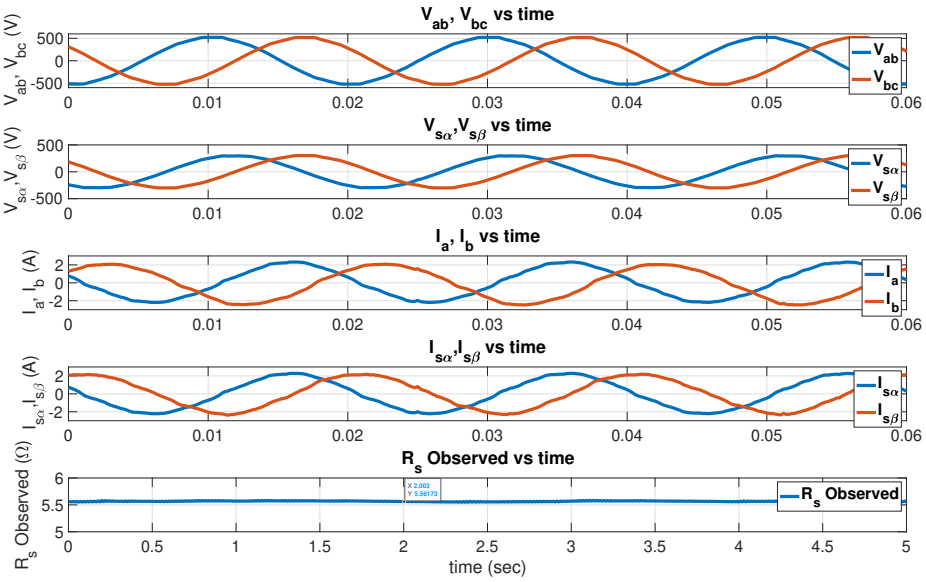


Figure 6.6: Variables used to estimate $\mathbf{R_s}$ at 3000 RPM with 1 Ω series resistance

6.2.4 $\mathbf{R_s}$ Value vs Speed Estimation Performance of Sensorless Vector-Controlled Induction Motor

The vector control algorithm implemented in this thesis is direct vector control. We have utilized the motor's voltage model to control the motor. The details of the algorithm are given in Section 2.4.11 and Section 2.4.16. To investigate the effect of these estimation errors on the speed estimation algorithm, we have simulated our vector control model shown in Figure B.11 such that the parameters of the test motor kept constant while the stator resistance R_s value in the control algorithm is changed between $\pm\%3$. Table 6.2 shows the simulation results of speed estimation performance for $R_s\pm\%2\mathrm{R}_s$ and $R_s\pm\%4\mathrm{R}_s$ values. Table 6.3 shows the error in the speed estimation for the given R_s values. In the induction motors the R_s value can rise up to $\%30$ percent and the IAEKF algorithm can detect this rise, with an error $\pm\%3$, and the effects of these errors on the sensorless vector-controlled induction motor is given in the following tables. The error is calculated by "% error=100x($\hat{\omega}_s$-ω_s)/ω_s" equation in Table 6.3, where $\hat{\omega}_s$ is the estimated speed, ω_s is the actual speed.

Table 6.2: Estimated Speed value vs R_s value at different speeds

R_s value vs Speed Estimation	$\hat{\omega}_s$ @ 62.8 rad/s	$\hat{\omega}_s$ @ 157 rad/s	$\hat{\omega}_s$ @ 314 rad/s
0.96*R_s	59.5 rad/s	155.8 rad/s	313 rad/s
0.98*R_s	60.6 rad/s	155.9 rad/s	312.8 rad/s
1.02*R_s	60.2 rad/s	155.9 rad/s	312.8 rad/s
1.04*R_s	60 rad/s	155.8 rad/s	313 rad/s

Table 6.3: Estimated speed error vs R_s at different speeds

R_s Value vs $\hat{\omega}_s$ Error	Error @ 62.8 rad/s	Error @ 157 rad/s	Error @ 314 rad/s
0.96*R_s	%5.25	%0.07	%0.03
0.98*R_s	%3.5	%0.07	%0.03
1.02*R_s	%4.1	%0.07	%0.03
1.04*R_s	%4.4	%0.07	%0.03

6.2.5 Stator Resistance Estimation Under Load Torque

The estimation performance of the motor is tested under load torque. The motor is started under no-load condition, after 6 seconds, the load motor has loaded the test motor with 3Nm load torque. The IAEKF estimator performs well under the load disturbance as can be seen in Figure 6.7. This experiment is conducted with window length as N=4 and diagonal elements of the noise covariance as $R_k[A^2] = 1$. The load motor and the test motor can be seen in Figure 6.8.

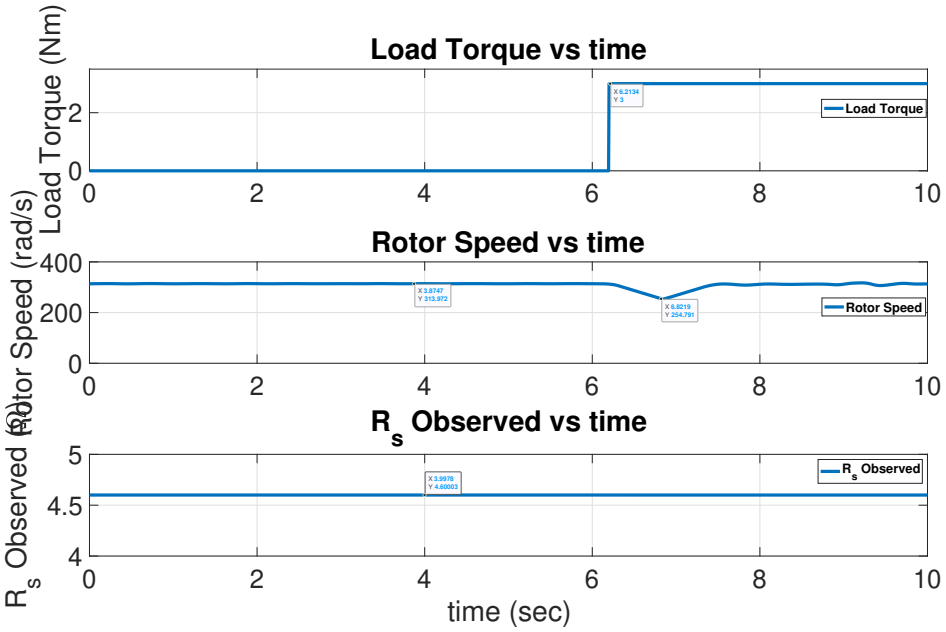


Figure 6.7: Stator resistance estimation under 3 Nm load torque

6.2.6 The Experimental Setup and the DSPACE ControlDesk Software

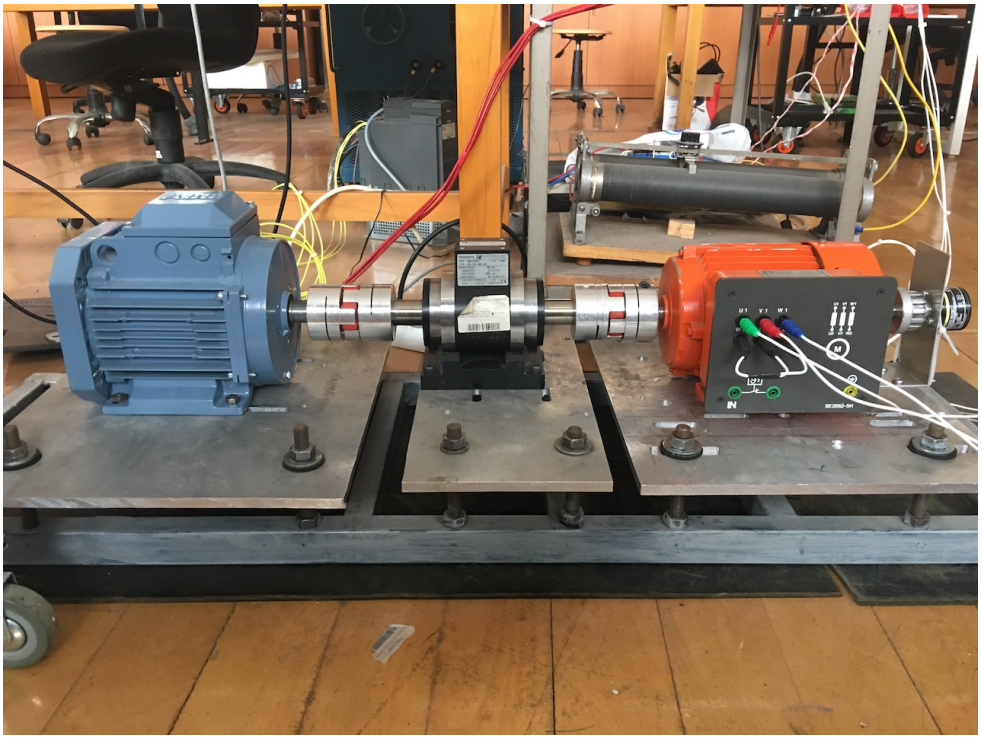


Figure 6.8: Experimental setup

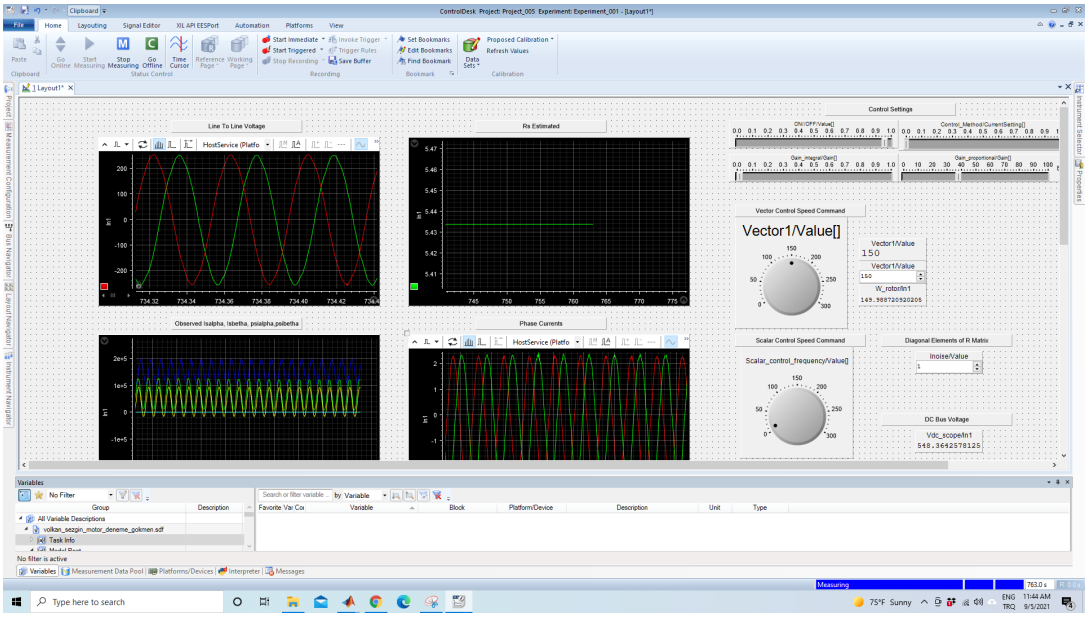


Figure 6.9: DS1104 ControlDesk software

The measurements and control in the experiments are done by the DSPACE ControlDesk software which is shown in Figure 6.9.

6.3 Conclusion

It is shown in this chapter that a novel IAEKF estimator may be used to estimate the induction motor's stator resistance R_s online while the motor is running. The IAEKF's estimate performance is tested with a vector-controlled real-time induction motor driver with experiments. The experiments are done with the custom inverter designed by Kayhan [3] to test the performance of the novel IAEKF estimator performance at different frequencies. The stator resistance R_s estimation is done at 10, 25, 50 Hz. The accuracy of the input parameter is crucial for IAEKF estimator's performance. The vector control algorithm predicts the speed with a much smaller error when this method is used. In these experiments the $R_k[A]$ values are different with respect to values used at the standstill tests, we have found these values with the trial and error method. The reason for different $R_k[A]$ values may be the noise we can not model in the running system. To conclude, the IAEKF algorithm is very useful for self-tuning and also for correcting the stator resistance under running conditions to

obtain more accurate control.

CHAPTER 7

CONCLUSION

In this thesis, the vector-controlled drive sofware is improved and verified. Due to the importance of determining stator resistance for self-tuning purpose a novel IAEKF stator resistance estimator is introduced. Firstly, simulation models are developed and improved. After verifying the simulation models, the models are embedded into the DSPACE DS1104 controller, and experiments are conducted in the real-time environment.

To improve the dynamic performance of the vector-controlled drive, parameters that affect the dynamic performance are identified by Kayhan [3], in his thesis work he mainly focused on DC link design and DC link compensation. In this thesis, we have focused on PI controller design, axis decoupling, and stator resistance estimation by the IAEKF algorithm. We have implemented the series PI controller firstly on the simulation environment, then in the DSPACE DS1104 controller. In the experiments, the series PI controller's advantage over conventional PI controller is observed. The series PI controller is designed by pole-zero cancellation method and it is implemented as described by [37]. In the pole-zero cancellation technique, the system stability is vulnerable to resistance value change caused by the temperature rise of the motor, to solve this issue, the resistance value in the DSP has to be updated. We have introduced a novel stator resistance estimator, implemented by IAEKF algorithm. And this algorithm is verified for running and standstill conditions. The algorithm's performance is tested for different parameters such as various window lengths, various diagonal elements of noise covariance matrix values and various frequencies. We got very good results from tests where the variable voltage transformer is utilized as power supply. Then we got also very good results with the PWM inverter tests. With

the utilization of the PWM inverter, we were able to test the algorithm's performance under various frequencies.

A rough calculation can be performed to demonstrate the proposed algorithm's mastery over the other Kalman filter-based prediction methods. In the prediction process of the adaptive fading-based EKF algorithm proposed in [32], for example, 10 subtractions, 197 additions, 253 multiplications, and 2 divisions are required. In the proposed IAEKF algorithm, the same process requires 10 subtractions, 14 additions, 38 multiplications, 4 divisions, and 2xN dimension registers. This simple analogy clearly demonstrates the proposed algorithm has much low calculation load. This algorithm can work parallel to the sensorless vector control algorithm to estimate the stator resistance value, however due to extra computational load, a powerful DSP have to be utilized.

In the future, we will work on estimating the other parameters of the induction motor, namely rotor resistance, magnetizing inductance, and leakage inductance. We will also renew the IGBTs with silicon carbide MOSFETs and change the controller with a much more computationally powerful controller.

REFERENCES

[1] K. Ogata, *Discrete-Time Control Systems (2nd Ed.)*. USA: Prentice-Hall, Inc., 1995.

[2] H. A. Toliyat, E. Levi, and M. Raina, "A Review of RFO Induction Motor Parameter Estimation Techniques," *IEEE Power Engineering Review*, vol. 22, no. 7, p. 52, 2002.

[3] M. Kayhan, "The effect of dc link filter design and compensation on dynamic performance of vector controlled drive," Master's thesis, Middle East Technical University, 2015.

[4] B. K. Bose, *Modern power electronics and AC drives* . Englewood Cliffs (N.J.) : Prentice-Hall, 2002.

[5] N. Mohan, *Vector Control of Induction-Motor Drives: A Qualitative Examination*. John Wiley Sons, Ltd, 2014.

[6] Texas Instruments, "User Guide: InstaSPIN-FOC and InstSPIN-MOTION," 2019.

[7] D. Willson, "Control structures." [online] Available: http://www.kappaiq.com [Accesed: September 2021].

[8] F.Blaschke, "The principle of field orientation as applied to the new transvector closed loop control system for rotating field machines," *Siemens Review*, vol. 34, pp. 217–220, 1972.

[9] K. Hasse, *Zur Dynamik drehzahlgeregelter Antriebe mit Stromrichtergespeisten Asynchronkurzschlufermotoren /.* PhD thesis, T. U. Darmstadt, 1969.

[10] S.-H. Kim, *Electric motor control : DC, AC, and BLDC motors* . Amsterdam, Netherlands : Elsevier, 2017.

[11] E. M. Committee, "IEEE Standard Test Procedure for Polyphase Induction Motors Having Liquid in the Magnetic Gap," *Test*, vol. 1995, 1995.

[12] M. Ruff and H. Grotstollen, "Off-line identification of the electrical parameters of an industrial servo drive system," *Conference Record - IAS Annual Meeting (IEEE Industry Applications Society)*, vol. 1, pp. 213–220, 1996.

[13] J. K. Seok, S. I. Moon, and S. K. Sul, "Induction machine parameter identification using PWM inverter at standstill," *IEEE Transactions on Energy Conversion*, vol. 12, no. 2, pp. 127–132, 1997.

[14] G. S. Buja, R. Menis, and M. I. Valla, "MRAS identification of the induction motor parameters in PWM inverter drives at standstill," *IECON Proceedings (Industrial Electronics Conference)*, vol. 2, pp. 1041–1047, 1995.

[15] C. B. Jacobina, J. E. Chaves Filho, and A. M. N. Lima, "Estimating the parameters of induction machines at standstill," *IEEE Transactions on Energy Conversion*, vol. 17, no. 1, pp. 85–89, 2002.

[16] J. R. Willis, G. J. Brock, and J. S. Edmonds, "A derivation of induction motor models from standstill frequency response tests," *IEEE Trans. Energy Conversion*, vol. 4, pp. 608–613, 1989.

[17] T. Kataoka, S. Toda, and Y. Sato, "On-line estimation of induction motor parameters by extended kalman filter," pp. 325–329 vol.4, 1993.

[18] L. C. Zai, C. L. DeMarco, and T. A. Lipo, "An Extended Kalman Filter Approach to Rotor Time Constant Measurement in PWM Induction Motor Drives," *IEEE Transactions on Industry Applications*, vol. 28, no. 1, pp. 96–104, 1992.

[19] J. Holtz and T. Thimm, "Identification of the machine parameters in a vector controlled induction motor drive," *Conference Record - IAS Annual Meeting (IEEE Industry Applications Society)*, no. pt 1, pp. 601–606, 1989.

[20] D. Cherifi, Y. Ouadi, and A. Tahri, "A luenberger state observer for stator resistance estimation in sensorless induction motor drives," *International Review on Modelling and Simulations*, vol. 6, pp. 360–369, Apr. 2013.

[21] M. Jouili, Y. Agrebi, Y. Koubaa, and M. Boussak, "A luenberger state observer for simultaneous estimation of speed and stator resistance in sensorless irfoc induction motor drives," in *2015 16th International Conference on Sciences and Techniques of Automatic Control and Computer Engineering (STA)*, pp. 898–904, 2015.

[22] D. Cherifi, "Online Stator and Rotor Resistance Estimation Scheme Using Sliding Mode Observer for Indirect Vector Controlled Speed Sensorless Induction Motor," *American Journal of Computer Science and Technology*, vol. 2, no. 1, p. 1, 2019.

[23] V. Vasic, S. Vukosavic, and E. Levi, "A stator resistance estimation scheme for speed sensorless rotor flux oriented induction motor drives," *IEEE Transactions on Energy Conversion*, vol. 18, no. 4, pp. 476–483, 2003.

[24] I.-J. Ha and S.-H. Lee, "An online identification method for both stator-and rotor resistances of induction motors without rotational transducers," *IEEE Transactions on Industrial Electronics*, vol. 47, no. 4, pp. 842–853, 2000.

[25] B. Karanayil, M. F. Rahman, and C. Grantham, "Online stator and rotor resistance estimation scheme using artificial neural networks for vector controlled speed sensorless induction motor drive," *IEEE Transactions on Industrial Electronics*, vol. 54, no. 1, pp. 167–176, 2007.

[26] B. Bose and N. Patel, "Quasi-fuzzy estimation of stator resistance of induction motor," *IEEE Transactions on Power Electronics*, vol. 13, no. 3, pp. 401–409, 1998.

[27] R. Demir, M. Barut, R. Yildiz, R. Inan, and E. Zerdali, "EKF based rotor and stator resistance estimations for direct torque control of induction motors," *Proceedings - 2017 International Conference on Optimization of Electrical and Electronic Equipment, OPTIM 2017 and 2017 Intl Aegean Conference on Electrical Machines and Power Electronics, ACEMP 2017*, pp. 376–381, 2017.

[28] R. Yildiz, M. Barut, E. Zerdali, R. Inan, and R. Demir, "Load torque and stator resistance estimations with unscented kalman filter for speed-sensorless control

of induction motors," *Proceedings - 2017 International Conference on Optimization of Electrical and Electronic Equipment, OPTIM 2017 and 2017 Intl Aegean Conference on Electrical Machines and Power Electronics, ACEMP 2017*, vol. 1, no. 1, pp. 456–461, 2017.

[29] M. Aydin, M. Gokasan, and S. Bogosyan, "Fuzzy based parameter tuning of EKF observers for sensorless control of Induction Motors," *2014 International Symposium on Power Electronics, Electrical Drives, Automation and Motion, SPEEDAM 2014*, pp. 1174–1179, 2014.

[30] J. Talla, Z. Peroutka, V. Blahnik, and L. Streit, "Rotor and stator resistance estimation of induction motor based on augmented ekf," in *2015 International Conference on Applied Electronics (AE)*, pp. 253–258, 2015.

[31] M. Jannati, S. Anbaran, D. M. Zaheri, N. R. N. Idris, and M. J. A. Aziz, "A new speed sensorless svm-dtc in induction motor by using ekf," in *2013 IEEE Student Conference on Research and Developement*, pp. 94–99, 2013.

[32] E. Zerdali, R. Yildiz, R. Inan, R. Demir, and M. Barut, "Adaptive Fading Extended Kalman Filter Based Speed-Sensorless Induction Motor Drive," *Proceedings - 2018 23rd International Conference on Electrical Machines, ICEM 2018*, pp. 1367–1373, 2018.

[33] Z. Yin, G. Li, Y. Zhang, J. Liu, X. Sun, and Y. Zhong, "A Speed and Flux Observer of Induction Motor Based on Extended Kalman Filter and Markov Chain," *IEEE Transactions on Power Electronics*, vol. 32, no. 9, pp. 7096–7117, 2017.

[34] E. Zerdali, "Adaptive Extended Kalman Filter for Speed-Sensorless Control of Induction Motors," *IEEE Transactions on Energy Conversion*, vol. 34, no. 2, pp. 789–800, 2019.

[35] A. Almagbile, J. Wang, and W. Ding, "Evaluating the Performances of Adaptive Kalman Filter Methods in GPS/INS Integration," *Journal of Global Positioning Systems*, vol. 9, no. 1, pp. 33–40, 2010.

[36] X. L. Li, J. G. Park, and H. B. Shin, "Comparison and evaluation of anti-windup PI controllers," *Journal of Power Electronics*, vol. 11, no. 1, pp. 45–50, 2011.

[37] D. Willson, "Teaching your pi controller to behave." [online] Available: https://e2e.ti.com/blogs [Accesed: September 2021].

[38] H. van der Broeck, H.-C. Skudelny, and G. Stanke, "Analysis and realization of a pulsewidth modulator based on voltage space vectors," *IEEE Transactions on Industry Applications*, vol. 24, no. 1, pp. 142–150, 1988.

[39] J. Orsinger, "Using the angle decoder modules." [online] Available: https://imperix.com/doc/help/using-the-angle-decoder-modules [Accesed: September 2021].

[40] "Ieee standard test procedure for polyphase induction motors and generators," *IEEE Std 112-2017 (Revision of IEEE Std 112-2004)*, pp. 1–115, 2018.

[41] G. Welch and G. Bishop, "An Introduction to the Kalman Filter," *In Practice*, vol. 7, no. 1, pp. 1–16, 2006.

[42] R. Demir and M. Barut, "Novel hybrid estimator based on model reference adaptive system and extended Kalman filter for speed-sensorless induction motor control," *Transactions of the Institute of Measurement and Control*, vol. 40, no. 13, pp. 3884–3898, 2018.

[43] M. Barut, R. Demir, E. Zerdali, and R. Inan, "Speed-sensorless direct torque control system using Bi-input extended Kalman filter for induction motors," *International Aegean Conference on Electrical Machines and Power Electronics, ACEMP 2011 and Electromotion 2011 Joint Conference*, no. September, pp. 343–346, 2011.

[44] E. Laroche, E. Sedda, and C. Durieu, "Methodological insights for online estimation of induction motor parameters," *IEEE Transactions on Control Systems Technology*, vol. 16, no. 5, pp. 1021–1028, 2008.

[45] M. Barut, S. Bogosyan, and M. Gokasan, "Speed-sensorless estimation for induction motors sing extended Kalman filters," *IEEE Transactions on Industrial Electronics*, vol. 54, no. 1, pp. 272–280, 2007.

[46] E. Monmasson, L. Idkhajine, M. N. Cirstea, I. Bahri, A. Tisan, and M. W. Naouar, "Fpgas in industrial control applications," *IEEE Transactions on Industrial Informatics*, vol. 7, no. 2, pp. 224–243, 2011.

[47] L. Idkhajine, E. Monmasson, and A. Maalouf, "Fully fpga-based sensorless control for ac drive using an extended kalman filter," in *2009 35th Annual Conference of IEEE Industrial Electronics*, pp. 2925–2930, 2009.

[48] I. M. Alsofyani and N. R. N. Idris, "Simple flux regulation for improving state estimation at very low and zero speed of a speed sensorless direct torque control of an induction motor," *IEEE Transactions on Power Electronics*, vol. 31, no. 4, pp. 3027–3035, 2016.

[49] I. M. Alsofyani and N. R. N. Idris, "Lookup-table-based dtc of induction machines with improved flux regulation and extended kalman filter state estimator at low-speed operation," *IEEE Transactions on Industrial Informatics*, vol. 12, no. 4, pp. 1412–1425, 2016.

[50] F. Auger, M. Hilairet, J. M. Guerrero, E. Monmasson, T. Orlowska-Kowalska, and S. Katsura, "Industrial applications of the kalman filter: A review," *IEEE Transactions on Industrial Electronics*, vol. 60, no. 12, pp. 5458–5471, 2013.

APPENDIX A

EXPERIMENTAL MOTOR PARAMETERS

Table A.1: Nameplate of the SE 2662-5H

Rated Voltage (V)	Rated Current (A)	Rated Power (kW)	Rated Speed (RPM)	IP
380	2.7	1	2830	20

Table A.2: Parameters of the SE 2662-5H

$R_s[\Omega]$	$R_r[\Omega]$	$L_m[H]$	$L_{ls}[H]$	$L_{lr}[H]$	$J_l[kg.m^2]$	$b_L[N.m/(rad/s)]$
4.501	6	0.375	0.0117	0.0117	0.00553	0.001

"Flux producing current" I_{sdref} must be adjusted to the value that corresponds to the non-saturated flux maximum. Following equations shows how to determine flux producing current value using the motor's nameplate data.

$$E_m = \sqrt{\left[(V_{1n}\cos\varphi_n - R_s I_{sn})^2 + (V_{1n}\sin\varphi_n - w_s\sigma L_s I_{sn})^2\right]}$$

$$\psi_m = \frac{E_m\sqrt{2}}{w_s}$$

$$I_{sdref} = \frac{\psi_m}{L_m}$$

Putting the nameplate values in these equations gives :

$$E_m = \sqrt{[(220 \cdot 0.76 - 4.35 \cdot 2.7)^2 + (220 \cdot 0.65 - 314 \cdot 0.0233 \cdot 2.7)^2]} = 198.4 \text{ V}$$

$$\psi_m = \frac{198.4 \cdot \sqrt{2}}{314} \cong 0.9Wb$$

$$I_{sdref} = \frac{0.9}{0.375} \cong 2.4 \text{ A}$$

APPENDIX B

VECTOR CONTROL SOFTWARE

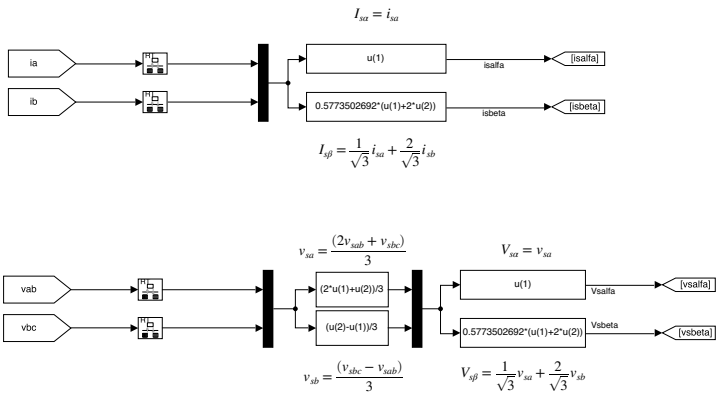


Figure B.1: Clarke transforms implemented in Simulink

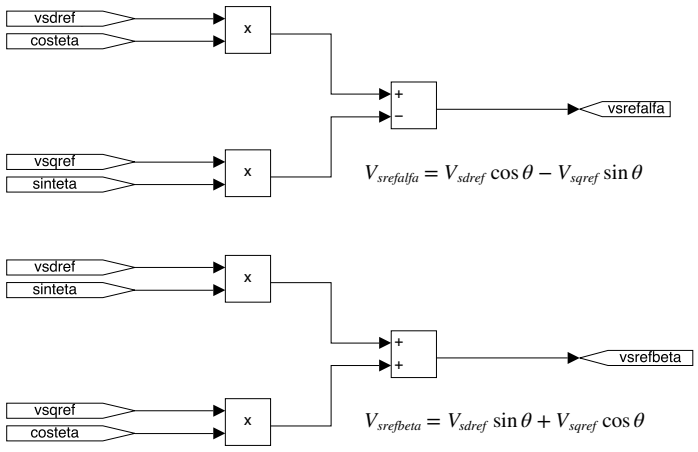


Figure B.2: Inverse Park transform implemented in Simulink

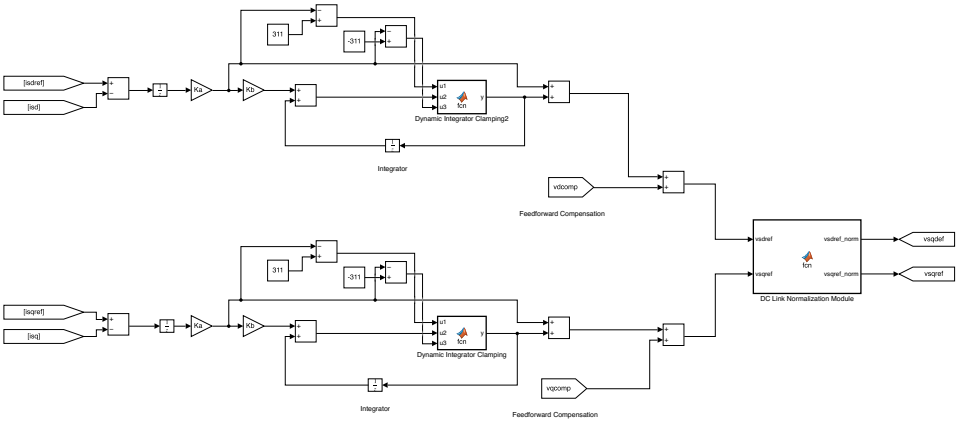


Figure B.3: Series PI controller implemented in Simulink

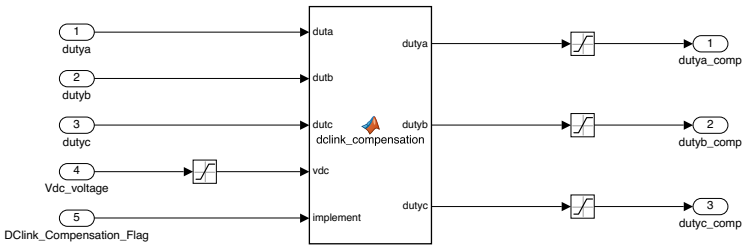


Figure B.4: The implementation of the DC link compensation in Simulink

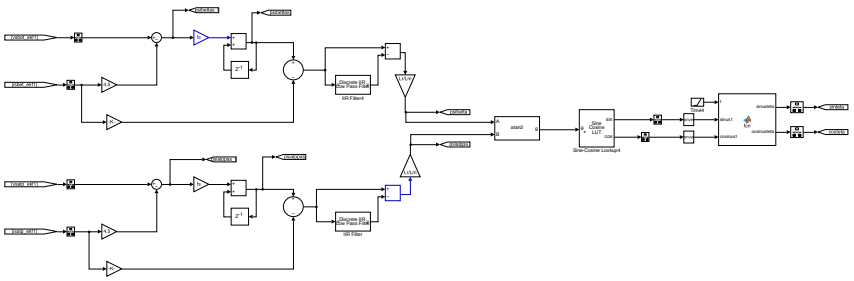
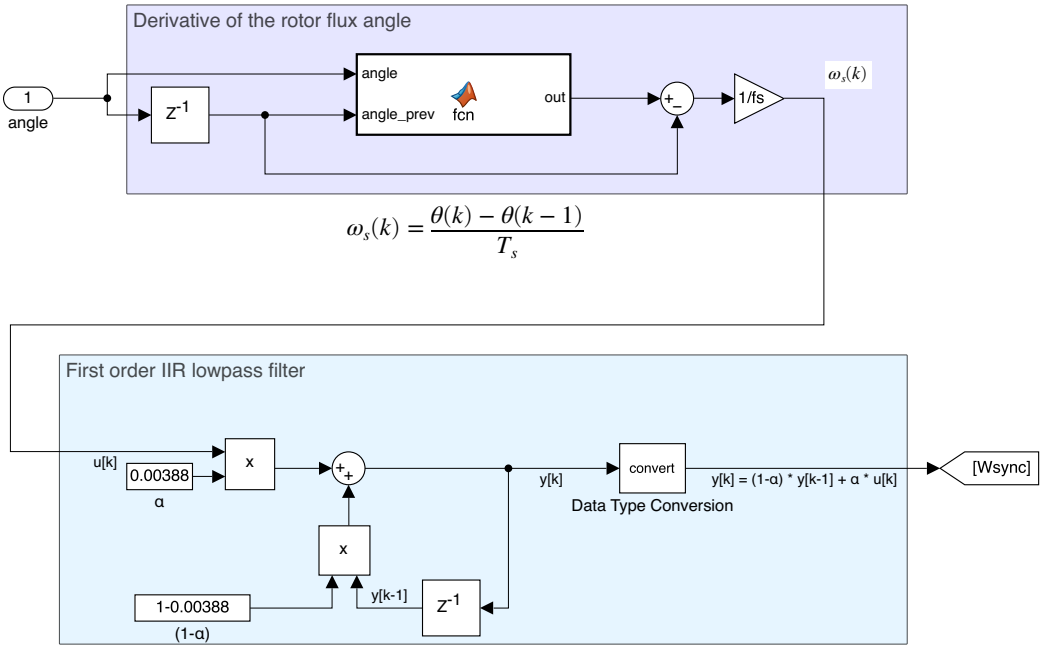


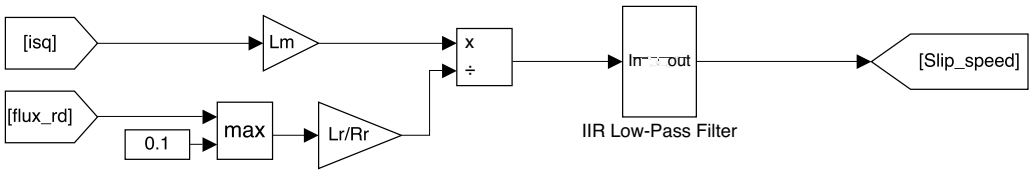
Figure B.5: Rotor flux observer



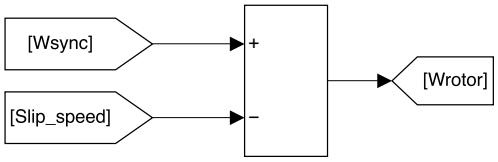
$$\omega_s(k) = \frac{\theta(k) - \theta(k-1)}{T_s}$$

$$H(z) = \frac{\alpha}{1 + (\alpha - 1)z^{-1}}$$

Figure B.6: Synchronous speed calculation and IIR filter



$$s * w_S = \frac{L_m}{\tau_r} * \frac{I_{sq}}{\lambda_{rd}}, \tau_r = \frac{L_r}{R_r}$$



$$w_r = w_S - s * w_S$$

Figure B.7: Slip speed and rotor speed calculation

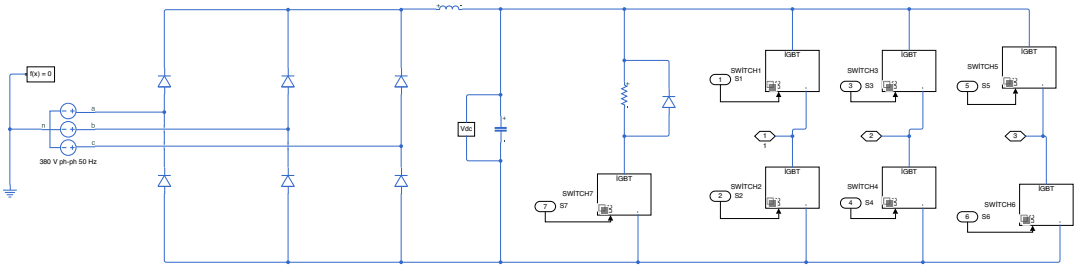


Figure B.8: IGBT inverter in Simulink

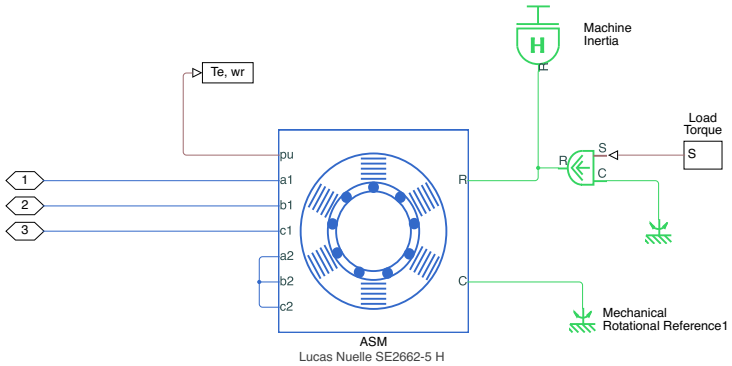


Figure B.9: Motor model

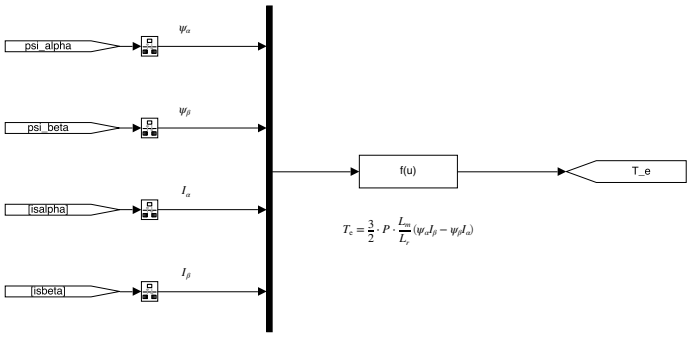


Figure B.10: Torque calculation in Simulink

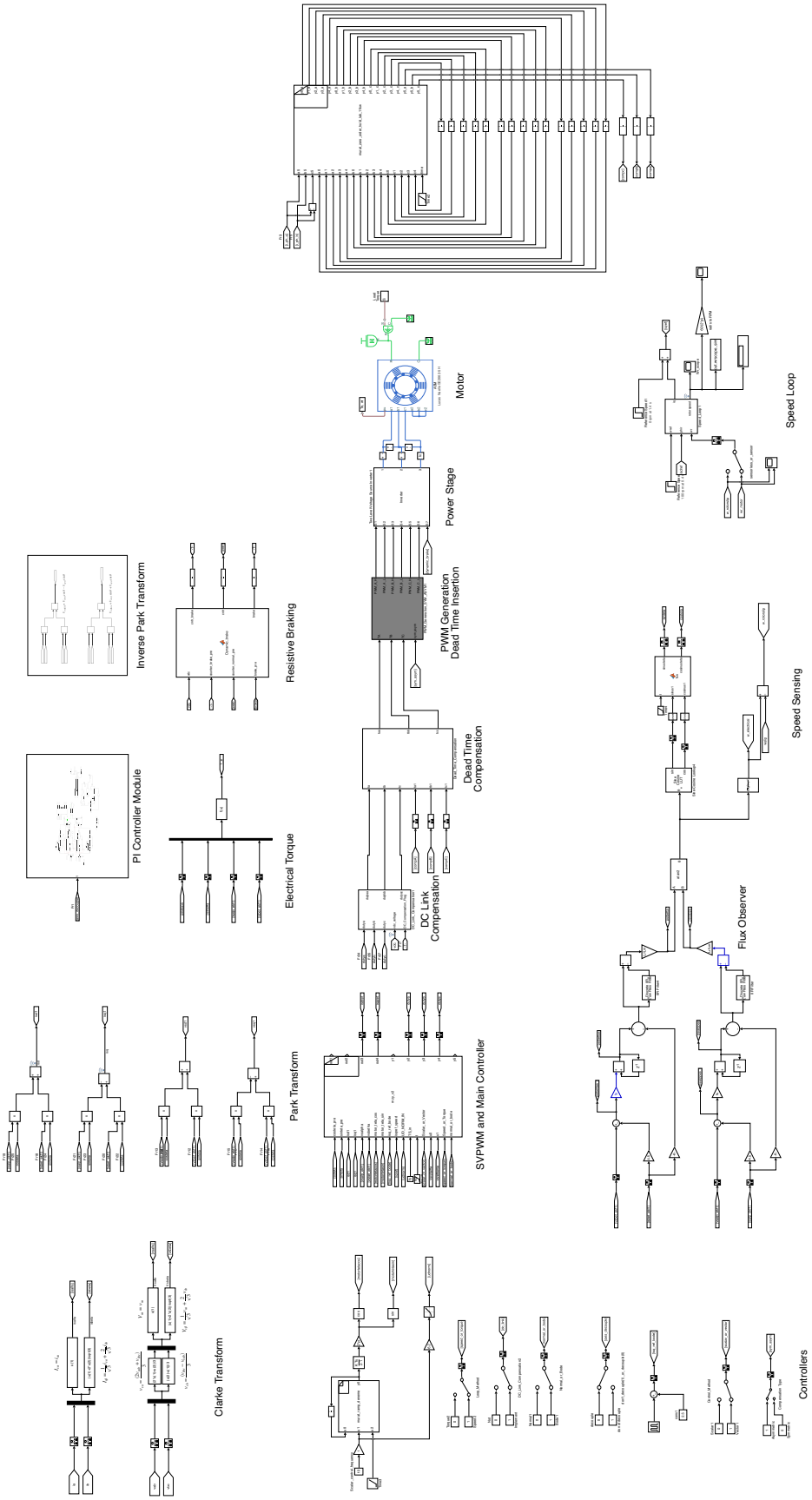


Figure B.11: Simulation file implemented in Simulink

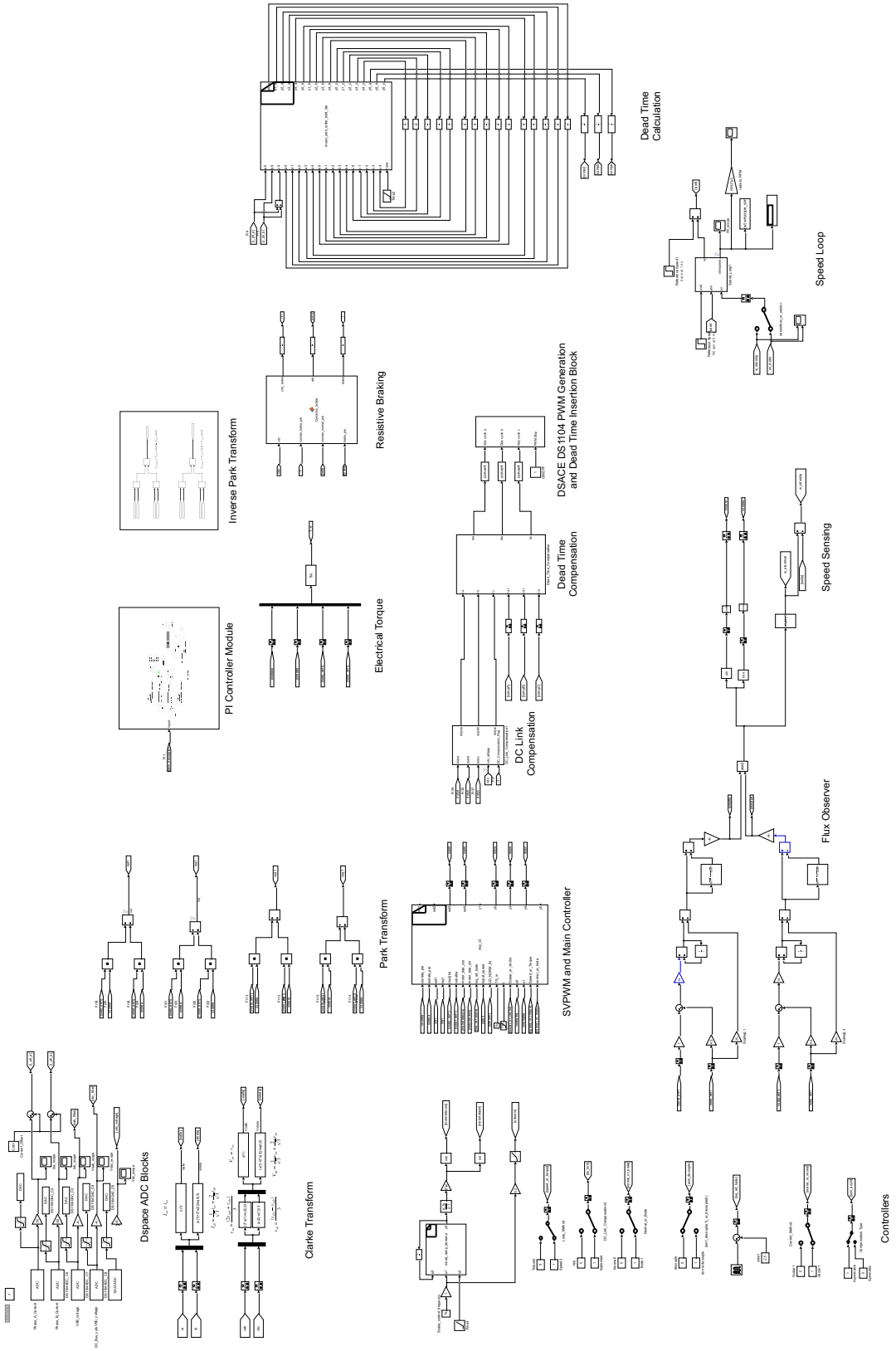


Figure B.12: Simulation file modified and embedded in the DSPACE DS1104